



Promoting Trust and Civility in On-line Interactions

AssertID, Inc.
Mill Valley, CA
www.assertid.com

June 28, 2013

Secretary of the Commission
Federal Trade Commission

600 Pennsylvania Avenue, N.W.
Washington, D.C. 20580

RE: Request for review and approval of AssertID's "verifiable parental consent" method under Part 312.12(a) of the Children's Online Privacy Protection Rule.

Dear Secretary:

Pursuant to the Children's Online Privacy Protection Act (16 CFR Part 312.12(a)), AssertID, Inc. ("AssertID") respectfully submits the following application requesting approval of AssertID's "*verifiable parental consent method*" as an "approved method" for obtaining verifiable parental consent under the COPPA Rule.

AssertID's verifiable parental consent method consist of the following 6 processes which collectively ensure compliance with Part 312.5(b)(1) of the COPPA Rule – ensuring that the individual granting parental consent (or revoking such consent previously granted) is in fact the parent of the child.

1. A process for parental notification of consent-request.
2. A process of presentment of consent-request direct notices to parents.
3. A process for recording and reporting a parent's response to a consent-request to the Operator.
4. A process for recording and reporting a parent's request to revoke consent previously granted and to have their child's personal information deleted.
5. A process of verification of the parent-child relationship.
6. A process to ensure that only a parent of the child for whom consent is being requested can access and respond to such requests.

This application is divided into the following four sections:

1. About AssertID
2. AssertID VPC Method Description
3. Analysis of AssertID's VPC Methods Compliance with Part 312.5(b)(1)
4. Exhibits

The Exhibits section consists of:

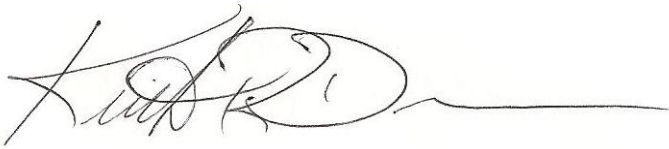
- A. AssertID™ Verification Technology - (proprietary & confidential – to be redacted from the public record)
- B. AssertID Patent Application “Method and System for On-line Identification Assertion” (by reference)

Because of its proprietary nature, we ask that Exhibit A “AssertID™ Verification Technology” be redacted from the public record.

For convenience, we have included with this application 2 CDs, each containing; digital copies of this application in both MS Word and PDF formats and a PDF copy of the referenced AssertID patent application “Method and System for On-line Identification Assertion”.

We look forward to working with the FTC during the public comment and evaluation process to answer any questions that might arise regarding our application, and if necessary to modify our application to enhance clarity or to provide additional information.

Sincerely,

A handwritten signature in black ink, appearing to read 'Keith Dennis', followed by a long horizontal line.

Keith Dennis
President, AssertID, Inc.

Contents

1	About AssertID.....	5
1.1	Introduction.....	5
1.2	Background.....	5
2	AssertID VPC Method Description	7
2.1	Definitions.....	7
2.2	Method Summary.....	7
2.3	<i>ConsentID™</i> Overview.....	8
2.4	Operator Portal	9
2.4.1	Administrator Account Creation	9
2.4.2	Domain Registration	10
2.4.3	Domain Verification.....	10
2.4.4	Application Policy Specification.....	10
2.4.5	Policy Validation	15
2.4.6	Application Registration	16
2.4.7	Application Integration	18
2.5	Consent-request Notification Process	19
2.5.1	Email Notification	19
2.5.2	Facebook Notification	20
2.5.3	Text-message Notification.....	20
2.5.4	Respond to Request (within Notification)	20
2.5.5	First Consent-request	21
2.6	Parent Portal.....	22
2.6.1	Consent-Request Presentment.....	22
2.6.2	Consent-request (direct notice) Presentment.....	22
2.6.3	Consent Revocation (delete child's PII) Process.....	25
2.7	AssertID Verification Process.....	27
2.7.1	Creating a User's AssertID.....	27
2.7.2	Building a User's Social-graph.....	28
2.7.3	Trust-score calculation.....	28
2.7.4	Verification Process	29

3	Analysis of AssertID’s VPC Method Compliance with Part 312.5(b)(1).....	31
3.1	Parent Identity Verification	32
3.2	Parent-child Relationship Verification.....	32
3.2.1	Social-graph Verification Method	32
3.2.2	Credit-card method	32
3.2.3	Susceptibility to Fraud.....	32
3.3	Restricted Access to Consent Mechanism	33
4	Exhibits	33
A.	AssertID Verification Technology.....	33
B.	AssertID Patent Application “Method and System for On-line Identification Assertion” (AssertID Patent Application).....	33

1 About AssertID

1.1 Introduction

AssertID, Inc. is a private for-profit corporation specializing in the development of proprietary privacy and identity verification technologies, products and services which leverage advances in the science of Social Network Analysis.

AssertID™ has developed patent-pending processes which, through a combination of peer-verifications and analysis of an individual's social-graph can derive a quantitative score ("*trust score*") which is a quantitative measure of the likelihood that an individual's self-asserted identity attributes are accurate.

AssertID's core identity-verification service allows participating users to create a digital identity credential (an "AssertID") which contains the user's self-asserted identity attributes (e.g. Name, Age, Gender, Email, Photo, Location, etc.) provided by the user. Once created, a user's AssertID becomes available for verification by select friends and family from the user's social-graph.

As an individual's AssertID is verified by their friends and family, the AssertID process analyses the number, quality and nature of these peer-verifications ("direct verifiers") and of these verifier's verifiers ("indirect verifiers"). From this analysis AssertID derives a numeric *trust score* which is a reliable indicator of the accuracy of the user's self-asserted attributes.

This *trust score* is dynamic, meaning that an individual's *trust score* is continuously updated as changes or additions are made to the identity attributes contained in an individual's AssertID credential and as additional peer verifications are performed. This identity-verification process forms the basis for AssertID's "verifiable parental consent" method.

1.2 Background

Upon review of the FTC's published NPRM in 2011, AssertID recognized an opportunity to leverage AssertID's core identity-verification technology to address the challenges website operators and application developers were facing in obtaining "verifiable parental consent" (now "VPC") to maintain compliance with the COPPA Rule. To better meet the needs of both operators and parents, AssertID developed a COPPA-compliant "verifiable parental consent" method, and incorporated that method into a web-service (now "*ConsentID*™") designed specifically to achieve the following five key objectives:

1. **Operational Simplicity:** *ConsentID*™ simplifies the consent-request process for operators. This is achieved by providing operators with a simple, web-based (fill-in-the-forms) self-registration process that guides the operator through the process of defining their website, application or service (now "Application") and their associated user information practices in a clear and concise manner.

Once this registration process is complete, an operator can initiate the VPC process with a simple, secure call to the *ConsentID*™ API. This call sets in motion a process of

presentment of the consent-request to the parent and return of the parent's decision back to the operator via the *ConsentID™* API.

2. **Clarity of Presentment:** All consent-request direct notices are presented to parents through the *ConsentID™* Parent Portal. Parents are directed to the portal by way of an email notification (and other optional notification methods) for each new consent-request. Parents can also access all pending requests, at any time, through the same *ConsentID™* Parent Portal.

Consent-requests are presented in a consistent, easily understood format designed specifically to provide parents with the information they need when they need it.

3. **Ease-of-Use for Parents:** The *ConsentID™* process strives to make the consent-request presentment, review and response process as simple and non-intrusive for parents as possible. AssertID's identity-verification technology obviates the need for credit-card transactions, printing, signing and faxing of consent forms or other cumbersome or process-intensive identity verification methods. *ConsentID™* is less intrusive because there is no need for parents to divulge sensitive personal or financial information. Because the need for sensitive personal information is diminished, so too is the risk of unintended disclosure of sensitive information.

The *ConsentID™* Parent Portal provides parents with a single password-protected interface where they can access and process all parental-consent requests from all participating Applications and for all of their children. This same portal allows parents to review consents previously granted and if desired to revoke that consent. This revocation effectively serves as a delete my child's PII request.

4. **Verification of the Parent/Child Relationship:** *ConsentID™* verifies that the individual granting consent is in fact the parent or guardian (the "*parent*") of the child for whom consent is being requested. *ConsentID™* achieves this by creating a unique digital credential (a "*ConsentID™*") for each unique parent-child pair. This credential contains at a minimum; the name of the parent and first-name of the child. This credential has its own *trust score* which represents the strength of the verification of the parent-child relationship it represents. The parent-child relationship in essence becomes another self-asserted attribute of the parent's AssertID and is verified in the same manner as the parent's other self-asserted attributes. As this parent-child relationship is verified by friends and family, the *trust score* of the *ConsentID™* increases. A minimum trust score of 7 is required before the *ConsentID™* is "enabled" allowing the parent to grant (or revoke) consent for this child.

The veracity of this verification is configurable within *ConsentID™* and can therefore be made as stringent as is deemed appropriate. We believe that even a relatively low veracity setting for this specific attribute will result in verification that the individual granting consent is in fact the parent that is significantly stronger than the currently approved methods.

5. **Low Cost:** In order to mitigate the “cost of compliance” burden for operators, AssertID will offer the basic *ConsentID™* service completely free of charge. Additional premium services will be offered on a fee-basis. *ConsentID™* is always free to end-users (parents).

In summary, AssertID’s objective with *ConsentID™* is to remove significant impediments to COPPA compliance faced by operators while at the same time empowering parents with the tools and information they need to manage their children’s online privacy.

More specifically, we believe that the *ConsentID™* service offers the considerable benefits of a “*sound and practical solution that will serve a broad base of operators.*” as characterized in the following excerpt from the final COPPA rule amendments.

“The Commission believes that common consent mechanisms, such as a platform, gaming console, or a COPPA safe harbor program, hold potential for the efficient administration of notice and consent for multiple operators. A well-designed common mechanism could benefit operators (especially smaller ones) and parents alike if it offers a proper means for providing notice and obtaining verifiable parental consent, as well as ongoing controls for parents to manage their children’s accounts.²³⁴ The Commission believes that such methods could greatly simplify operators’ and parents’ abilities to protect children’s privacy.”

2 AssertID VPC Method Description

2.1 Definitions

The terms “Operator”, “personally identifiable information (PII)”, “child” and “parent” as used in this document take their meaning from the same terms as defined in the Children’s Online Privacy Protection Act (COPPA) and its implementing Rule, 16 C.F.R. Part 312.

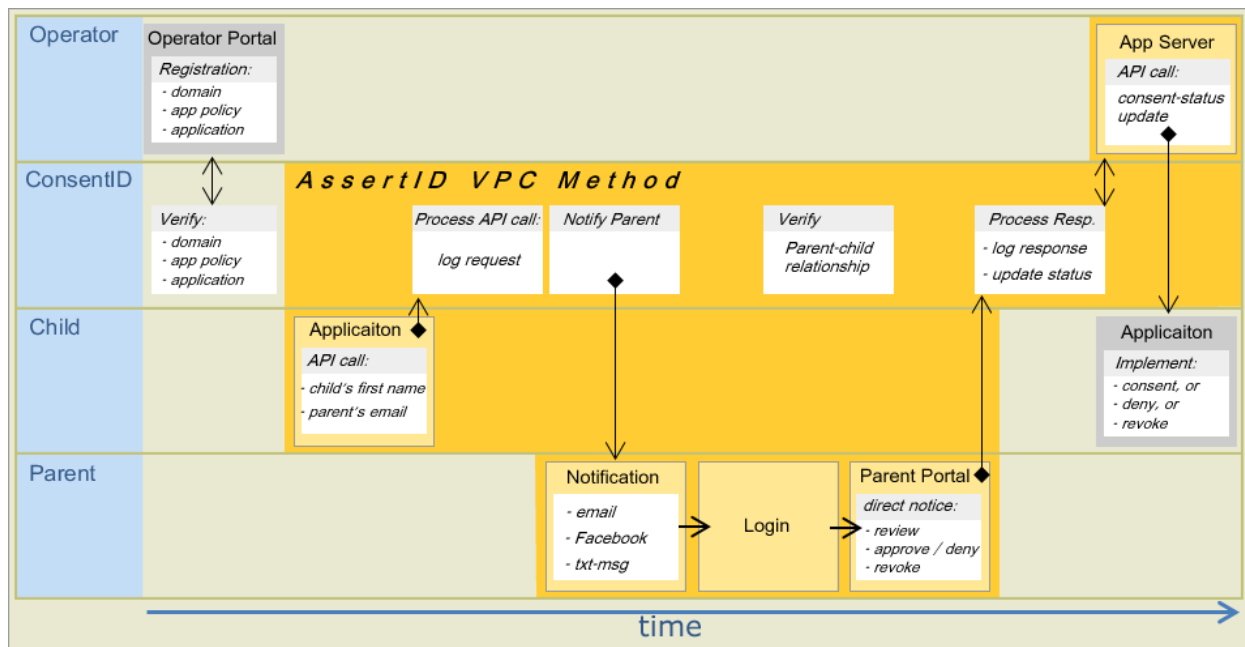
The term “Application” as used in this document is synonymous with and is used as a substitute for the phrase “*website or online service directed to children*” as defined in the Children’s Online Privacy Protection Act (COPPA) and its implementing Rule, 16 C.F.R. Part 312.

2.2 Method Summary

The VPC method for which AssertID is requesting approval consists of multiple components which collectively ensure compliance with Part 312.5(b)(1) of the COPPA Rule. These components (processes) which constitute the method for which AssertID seeks FTC approval are:

1. A process for parental notification of consent-request.
2. A process of presentment of consent-request direct notices to parents.
3. A process for recording and reporting a parent’s response to a consent-request to the Operator.

4. A process for recording and reporting a parent's request to revoke consent previously granted and to have their child's personal information deleted.
5. A process of verification of the parent-child relationship.
6. A process to ensure that only a parent of the child for whom consent is being requested can access and respond to such requests.



VPC Method Process-flow

As a practical matter, the VPC method for which AssertID is requesting approval would be difficult to evaluate absent a reference implementation of that method. For the purposes of this application, AssertID submits our *ConsentID™* service as the reference implementation of the VPC method for which we seek approval.

2.3 *ConsentID™* Overview

The *ConsentID™* service guides participating website operators and application developers (now “Operators”) through the entire VPC process. This consent-request process is preceded by the registration of each Operator, as well as each Application an Operator will make available to children under age 13.

In order to ensure full accountability and to protect against the possibility of unauthorized access of the *ConsentID™* API, every registered Operator and Application must have an associated Domain, and this Domain must be registered and validated before API calls are accepted from any Application hosted on this Domain.

Operators are guided through the specification of their information usage practices which are captured, validated and codified in the form of one or more Application policies (now “Application Policy” or “Application Policies”). Each registered Application must be associated with a valid Application Policy before consent-requests can be generated for that Application.

Operators are required to accept the *ConsentID™* terms-of-service (“TOS”), and in so doing agree to be legally bound by these TOS and to process all notifications, status changes and requests issued to them through the *ConsentID™* API in accordance with these TOS.

Upon completion of all registration steps, an Operator is provided with access to the *ConsentID™* API. Then, using a parent’s contact information, an Operator can initiate the parental-consent request process with a simple call to the *ConsentID™* API from within their Application’s user onboarding process.

Parents are notified of new consent-requests using the contact information provided by the parent or child. These notifications will direct the parent to the *ConsentID™* Parent Portal where the parent will be presented with all essential direct notice information as required under the COPPA Rule in a standardized and easily understood format. The parent is then guided through the consent-request review and response process.

After having reviewed the Application’s user information practices the parent can choose to decline or grant consent for the requesting Application.

Using this same Parent Portal, parents also have the ability to revoke consent(s) previously granted for any participating Application – effectively a “delete child’s PII” request.

The parent’s choices are communicated back to the Operator through the *ConsentID™* API. All significant events are time/date stamped and logged for audit and reporting purposes.

The *ConsentID™* Parent Portal provides parents with a single, password protected interface where they can manage and review all current and past consent-requests, for all of their children and for all participating Applications.

In addition, through this same Parent Portal, parents can find other Applications of possible interest to their children, review the information privacy practices for these Applications, and if desired pre-approve (grant consent) for these applications.

2.4 Operator Portal

The *ConsentID™* Operator Portal is a web-based administrative interface through which Operators can self-register and manage all of their Domains, Applications and Privacy Policies. Only those Applications properly registered through the Operator Portal can access the *ConsentID™* API.

2.4.1 Administrator Account Creation

Operators wishing to use *ConsentID™* and to access the Operator Portal are required to first create a password-protected administrative account through which all registration processes are

performed. Once created, an Operator can add additional administrators to this account. Each unique administrator must create their own user-id and password and indicate that they have read, accept and are bound by the *ConsentID™* TOS.

2.4.2 Domain Registration

An Operator must register each domain where Applications will be hosted. An Operator may register as many domains as are necessary to host their Applications.

2.4.3 Domain Verification

Each registered domain is verified to ensure that the Operator registering the domain is in fact the owner of the domain. To accomplish this verification, the Operator is issued a verification key that is unique to each registered domain. This verification key must be placed in a designated verification file and this file must be installed in the root directory of the registered domain. (Only an individual with authorized access to the domain server would be able to install this file.) Once installed, the operator initiates (through the Operator Portal) the verification of the newly registered domain by the *ConsentID™* server. Upon successful verification of the verification key, the domain is marked as “enabled” and registered Applications hosted on this domain can issue calls to the *ConsentID™* API.

2.4.4 Application Policy Specification

Registered Operators are guided through the specification of “Application Policies” designed to provide the detailed information collection and usage practices most critical to parents.

These Application Policies are distinct from the Operator’s “General Policy” which must be accessible from the Operators website or Application and which is more comprehensive in nature typically addressing an Operator’s complete information usage practices as well as providing the Operator’s contact and additional information as required under Part 312.4 of the COPPA Rule.

Application Policies are specific to one or more Applications and are designed to capture and communicate those information practices specifically required for inclusion in direct notices to parents under the COPPA Rule.

The Operator assumes all responsibility to ensure that their Application Policies are consistent with their General Policy.

An Operator may define as many Application Policies as are necessary to represent the information usage practices of each Application that will use the *ConsentID™* API.

As per the *ConsentID™* TOS, the Operator accepts full responsibility for the accuracy of all representations made in their Application Policies.

2.4.4.1 Create New Application Policy

To create a new Application Policy an Operator must provide a name for the Policy and the URL of the Operator’s General Policy, a link to which is displayed in each direct notice to parents.

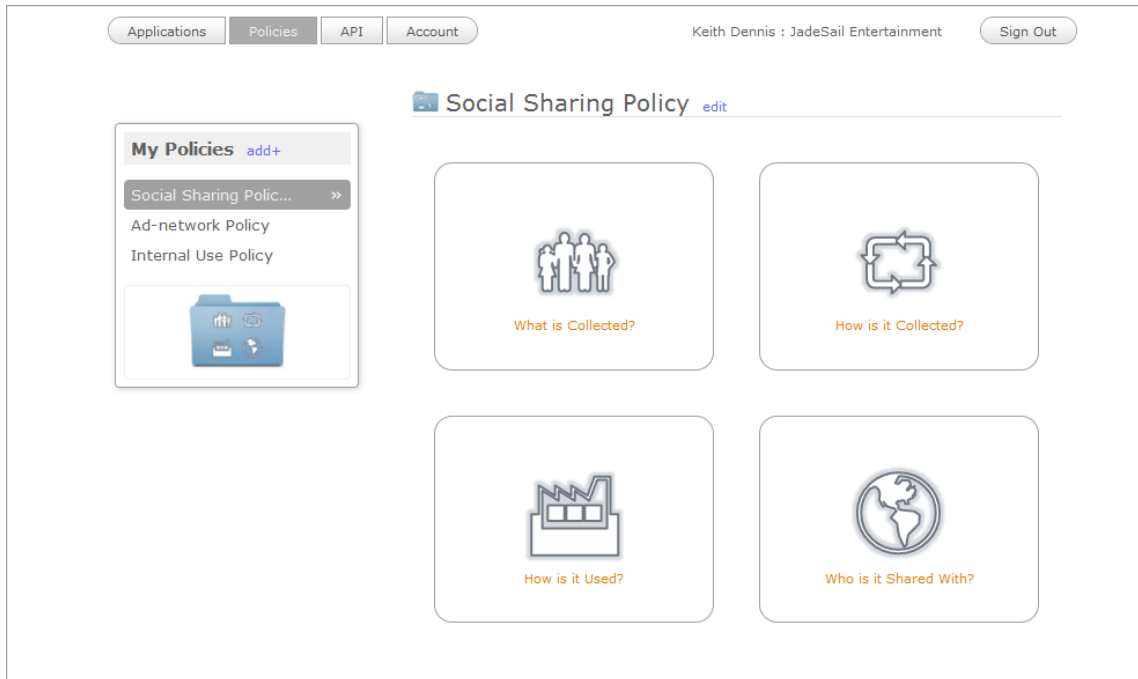
In addition, the Operator may optionally provide a “Policy Brief” - a simple text explanation of why the Operator is collecting personal information and how that information is used. This (optional) Policy Brief is intended as a means for the Operator to communicate their intent (in their own words) to the parent within the direct-notification.

The screenshot shows a web interface for managing policies. At the top, there are tabs for 'Applications', 'Policies', 'API', and 'Account'. The user is logged in as 'Keith Dennis : JadeSail Entertainment' and can 'Sign Out'. The main heading is 'Social Sharing Policy' with a 'delete' link. Below this, there is a folder icon and a prompt to 'Enter a new policy name below.'. The form fields are: 'Name' (Social Sharing Policy), 'Policy URL' (http://www.assertid.com/policies/privacy.htm), and 'Policy Brief' (In order for us to offer your child a full and engaging experience we need to allow your child to share with other children and family members using our service.). An 'Update' button is at the bottom.

Create New Policy Screen

The Application Policy specification process presents the Operator with a series of detailed questions related to their information usage practices. Questions are segmented (by screen) into four categories:

1. What data is collected? - (*Data screen*)
2. How is this data collected? - (*Process screen*)
3. How is this data used? - (*Usage screen*)
4. Who is this data shared with? - (*Sharing screen*)



Policies Tab

Screen-shots of the policy specification questions are provided below. Most check box line-items are self explanatory. For those items for which an Operator might require additional guidance, a hover-state informational message is displayed to provide such guidance.

Selected items display in bold with a solid check-box. Unselected items display as grey with an unfilled check-box. Line-items can be selected or unselected with a simple click.

In the sample Data screen below the “Contact info” line-item is in the hover-state - an informational message is displayed below the line-items and above the “Save Policy” button. Similar “guidance” is provided for other line-items when appropriate.

2.4.4.2 Data – (What data is collected?)

Domains Policies API Account Binko Caruso : AssertID, Inc. Sign Out

Policy 1

(DATA)

Given parental consent, what information about the child will be collected?

Personal Information:

- ☒ Name
- ☒ Physical address
- ☒ Photo, video, audio
- ☒ Parent's contact info
- ☒ Contact info
- ☒ Geolocation data
- ☒ Age
- ☐ Preferences, hobbies
- ☒ Phone number
- ☐ SSN
- ☒ Gender
- ☐ Other personal data

Persistent Identifiers & Behavioral Data:

- ☒ IP address
- ☐ Other identifier
- ☒ Other behavioral data
- ☐ Screen name
- ☒ Websites visited
- ☒ Device identifier
- ☐ Location tracking

Contact info includes any online identity that can be used to directly contact the child such as email address, screen name, instant messaging ID, social network ID, etc.

Save Policy Cancel

Information Collected Questionnaire


Data types are segmented into two categories for clarity:

1. Personal Information
2. Persistent Identifiers & Behavioral Data





2.4.4.3 Process – (How is data collected?)

DomainsPoliciesAPIAccount

Binko Caruso : AssertID, Inc. Sign Out



Policy 1



(PROCESS)

How is the information gathered?

☒ directly from the child

☐ from the parent

☒ from the session

☒ from the device

☐ from 3rd party databases

☐ from other sources


Save PolicyCancel

How Collected Questionnaire





2.4.4.4 Usage – (How is data used?)

DomainsPoliciesAPIAccount

Binko Caruso : AssertID, Inc. Sign Out



Policy 1



(USAGE)

What is the information used for?

☒ to contact the child

☒ to personalize the child's user experience

☒ to customize advertisements

☐ to enable social networking

☐ to perform behavioral analysis

Save PolicyCancel

Information Usage Questionnaire

2.4.4.5 Sharing – (Who is data shared with?)

Domains Policies API Account Binko Caruso : AssertID, Inc. Sign Out

Policy 1

(SHARING)

Who is the child's personal information shared with?

- ☒ the child's network of friends
- ☒ 3rd marketers and advertisers
- ☒ other 3rd parties
- ☐ the information IS NOT shared

Save Policy Cancel

Information Sharing Questionnaire

2.4.5 Policy Validation

As an aide to Operators, each privacy policy specification is analyzed whenever the “Save Policy” button is clicked to check for incomplete or inconsistent policies.

2.4.5.1 Incomplete Policy

An incomplete policy is any policy for which answers have not been provided for each of the four information usage practices categories.

2.4.5.2 Inconsistent Policy

An inconsistent policy is any policy for which assertions provided in one of the four information usage practices categories are inconsistent with one or more assertions made in another category.

For example - were an Operator to indicate on the *Data screen* that no personal information is collected, and then indicate on the *Sharing screen* that information is shared with 3rd parties, this represents an inconsistency in the policy specification. When detected, the Operator is notified of the omission or inconsistency and is then guided in correcting the policy specification. All omission and inconsistencies must be resolved before a Policy is enabled.

2.4.6 Application Registration

Application registration is the process by which an Operator provides all of the information necessary for *ConsentID™* to formulate consent-requests and their associated direct notices for presentment to parents.

2.4.6.1 Applications Tab

The *Applications* tab lists all Domains defined for an Operator and all Applications defined within each Domain. Through this screen the Operator can add new Applications and Domains or edit existing Applications and Domains.



Applications Tab screen-shot with three sample Applications within a single Domain (Jadesail.com)


2.4.6.2 Edit Application

The *Edit Application* screen (below) provides a simple means for an Operator to record all relevant information about an Application. This information when combined with an Application Policy provides the input necessary for direct notices to parents. The information captured includes:

1. **Name** – Application name
2. **Type** – Application type (selection list)
 - a. website
 - b. application
 - c. mobile-application
 - d. service
 - e. social Network
3. **Age Range** – intended age-range for this Application

4. **Description** – A short description of the Application to be presented as part of the direct notices.
5. **Policy** - (choose from list of defined Application Policies)
6. **Sharing:** (check box) - indicates if a non-sharing version of this application is supported. Includes a text-box to accommodate an explanation of the limits of the non-sharing version for presentation to the parent.
7. **Purchases** (check box) – used to indicate if this Application supports in-app purchases.
8. **Weblinks** (check box) – used to indicate if the Application contains links to external sites.
9. **Home Page:** - URL of the Application's or Operator's home-page
10. **About Page:** - URL of the Application's or Operator's about-page
11. **Contact Page:** - URL of the Operator's contact-page

Applications Policies API Account Binko Caruso : Mobile Apps, Inc. Sign Out



App: 829740251
ACTIVE

App bookworms delete

Edit application settings for *bookworms*.

Name:

bookworms

Type:

Mobile Application

Age Range:

3 to 14

Descr:

Where will knowledge take you? Discuss your favorite books with friends. There is a world to discover through reading.

Policy:

Policy 1

Sharing:

☒ This app supports a non-sharing mode

Choosing the non-sharing version of this app will exclude your child from receiving promotional offers for free or discounted e-books.

Purchases:

☐ This app facilitates online purchases

Weblinks:

☐ This app contains links to external sites

Home Page:

http://www.assertid.com

About Page:

https://www.assertid.com/about-us/what-we-do/

Contact Page:

http://www.assertid.com/policies/contact

Save Changes

Edit Application screen-shot

2.4.6.3 API Tab

The *API tab* allows the Operator to perform two tasks:

1. To easily generate a test API call for any properly defined Application in order to verify that all information is properly presented in the associated direct notice. If the Domain has not yet been verified or if the Application definition or Application Policy are incomplete, and error will be returned.

If everything is in order a consent-request will be generated and sent to the email address specified. The API call and response are displayed as an aid to the Operator in understanding the interface and in diagnosing any problems that might occur.

2. To initiate the Domain verification process - assuming the Domain verification file has been properly installed in the root directory; the Domain will be verified and marked as active.

The screenshot shows the 'API' tab selected in the top navigation bar. The user is logged in as 'Keith Dennis : JadeSail Entertainment' and can click 'Sign Out'. On the left, a sidebar contains 'ConsentID™ API', 'Consent Request (test)' (selected), and 'Verify Domain Name'. The main area is titled 'Consent Request (test)' and contains instructions: 'Select an application and enter a child's name to send a test Consent Request.' Below this are input fields for 'Application' (set to 'bookworms'), 'Parent Email' (redacted), and 'Child Name' (set to 'Lazar'). A 'Send Request' button is present. Below the form, the API call details are displayed: a POST request to 'https://www.assertid.com/api/request/consent' with data including app_id, secret, parent_email, and child_name. The JSON response is also shown, indicating a successful 'AFFIRMATIVE_CONSENT' status.

API Tab screen-shot with response from API test displayed

2.4.7 Application Integration

Upon completion of all registration processes (Operator, Domain & Application) the Operator may initiate the consent-request process with a call to the *ConsentID™* API. API calls to the *ConsentID™* server are secured using SSL.

2.4.7.1 Consent-request Process

The consent request process is initiated when a child expresses interest in an Application. This might be triggered by the child visiting the subject website or by downloading a mobile-application; essentially, any use-case that will trigger an *Operator's* on-boarding process for a

child under age 13. It is the *Operator*'s responsibility to make this determination and to ensure that this consent-request process is initiated previous to the collection of PII from the child in accordance with the COPPA Rule.

If the *Operator* determines that parental-consent is required they must collect certain information from the child in order to initiate this process.

To initiate a consent-request the *Operator* must obtain:

1. Parent's email address, and
2. first name of the child

This information is presented to *ConsentID™* via a call to the API which is then associated with the requesting Application. The Application registration information when combined with the information in the API call provides all of the information required for *ConsentID™* to notify the parent and present a consent-request to the parent in the form of a direct notice.

The parent's response to a request is date/time stamped, logged and communicated back to the Operator through the *ConsentID™* API.

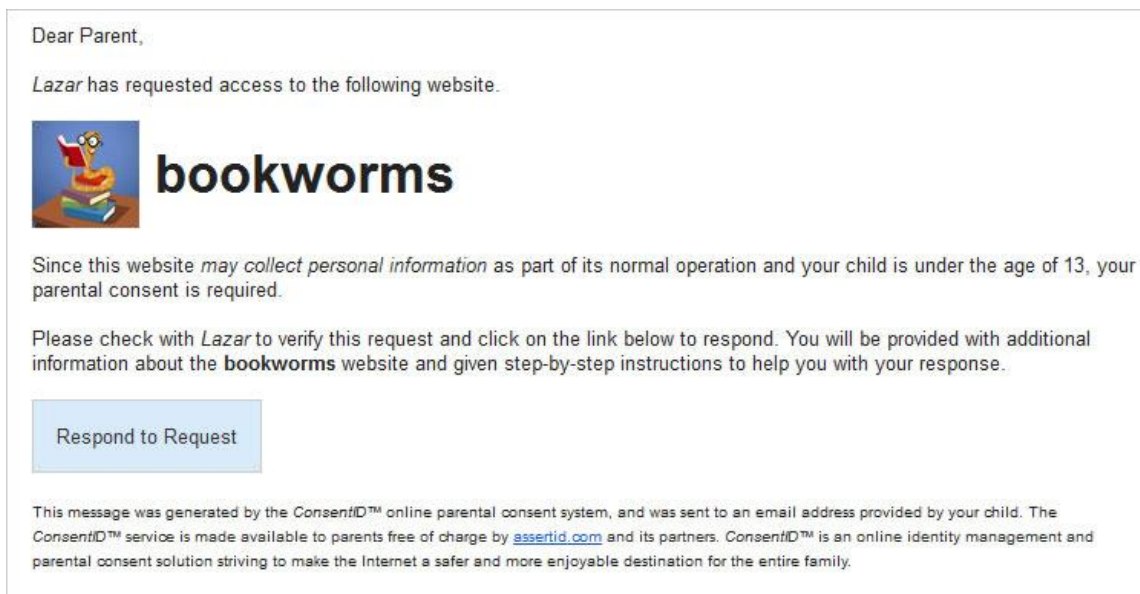
2.5 Consent-request Notification Process

ConsentID™ supports multiple notification methods in order to speed the consent-response process and to minimize the possibility of "missed" consent-requests. This approach also provides parents with maximum flexibility regarding how and where they wish to receive such notifications.

All notifications, regardless of delivery method, will contain a hyperlink that will direct the parent to the *ConsentID™* Parent Portal for presentment of the direct notice.

2.5.1 Email Notification

In all cases, an email notification is sent to the email address provided in the API call from within an Application's onboarding process. This email is personalized using the first name of the child for whom the request applies. This email notification contains the name of the requesting Application, an explanation of the nature of the request and a "Respond to Request" link.



Sample email notification for “bookworms” Application

2.5.2 Facebook Notification

Those parents who have a Facebook account and who have authorized the AssertID application will receive consent-request notifications within their Facebook account. Facebook currently posts such “application notifications” to the Facebook application dashboard. These application notifications are presented both within Facebook’s browser-based interface as well as within Facebook’s native mobile applications on both Android and iOS devices.

2.5.3 Text-message Notification

Parents may (optionally) choose to receive consent-request notifications via text-message. These notifications are not a substitute for the notifications sent to the parent’s email address rather, these notifications are a convenience feature available to parents who wish to receive them.

To receive text-message notifications, all of the following must be true:

1. The parent must indicate in their *ConsentID™* user profile that they wish to receive text-message notifications.
2. The parent must provide a valid mobile number where these text-messages are to be sent.
3. The parent must enter their login credentials to effect this change to their profile.

2.5.4 Respond to Request (within Notification)

All notifications, regardless of delivery method, contain a “Respond to Request” or equivalent hyperlink. Upon clicking the “Respond to Request” link in any notification, the parent is taken to the *ConsentID™* Parent Portal login process. The following use-cases are processed as described:

1. Parent has a *ConsentID™* account (and the email address matches the one associated with the parent's account) – parent is brought directly to the *ConsentID™* Parent Portal,
2. Parent has a *ConsentID™* account (and the email address does not match the one associated with the account) – if parent confirms that the consent request is valid (is from their child) then the new email address is added to their *ConsentID™* account, otherwise the consent-request is deemed invalid and is deleted.
3. Parent does not have a *ConsentID™* account – the parent is guided through account creation, completion of their AssertID, and the automatic generation of a *ConsentID™* representing this parent-child pair. Once completed the parent is taken to the *ConsentID™* Parent Portal.

2.5.5 First Consent-request

When a parent receives their first *ConsentID™* consent-request, the parent is guided through the creation of their *AssertID™* account and creation of their AssertID digital credential. In addition, a *ConsentID™* credential is automatically created representing the parent-child relationship for the child for whom the consent-request was issued.

2.5.5.1 AssertID creation

A parent's AssertID contains those self-asserted identity attributes that the parent is willing to share and have verified by close friends and family.

Currently, the possible self-asserted attributes contained on an AssertID include:

1. Photo
2. Full name
3. Gender
4. Age (represented as an actual age or an age-range at the AssertID owner's discretion)
5. Location

For user convenience, and provided the user has a Facebook account and authorizes the AssertID application, initial values for these identity attributes are taken from the user's Facebook profile. The user is encouraged to ensure the accuracy of all attributes on their AssertID and if necessary to modify them before requesting verification.

If a parent does not have a Facebook account, all identity attributes must be input by the parent. The self-asserted attributes contained in an AssertID are separate and distinct from those contained in a user's Facebook profile.

2.5.5.2 ConsentID Creation

A newly created *ConsentID™* initially contains the parent's full name and photo (if obtained from the parent's AssertID) and the child's first name (obtained from the API call). The following additional identity attributes may be present if provided by the parent:

1. Child's photo

2. Child's age (if birth date provided by parent)
3. Child's gender (if provided by parent)
4. Parent's age or age-range (if present on their AssertID)
5. Parent's photo (if present on their AssertID)

All identity attributes are editable by the parent.

Each ConsentID has a *trust score* indicating the veracity of the verification of this parent-child relationship. A minimal trust score of 7 out of 10 (7/10) is required before a *ConsentID™* is "enabled" allowing the parent to grant (or deny) parental-consent.

A *trust score* of 7/10 is only achievable when both the identity of the parent and the parent-child relationship have been verified by AssertID's proprietary verification process.

2.6 Parent Portal

The Parent Portal serves as the parent's primary interface to the *ConsentID™* service. This Portal provides parents with a single, secure interface for the administration of all parental-consent for all of their children and for all participating Applications.

All notifications, regardless of how they are delivered, direct the parent to the (password protected) Parent Portal to process the consent-request and associated direct notice. This ensures that only the parent can access these requests and that all requests are presented in a consistent manner.

2.6.1 Consent-Request Presentment

All valid consent-requests are posted to the parent's *ConsentID™* inbox and are only accessible through the password-protected Parent Portal.

2.6.2 Consent-request (direct notice) Presentment

When directed to the Parent Portal from a notification, the parent is immediately presented with the associated consent-request direct-notice.

Upon entering the Parent Portal (independent of a notification) parents are presented with all outstanding consent-requests on the landing "Home" screen. The left-hand column of this screen lists all outstanding consent-requests and the right-hand field contains the direct notice for the request currently selected on the left.

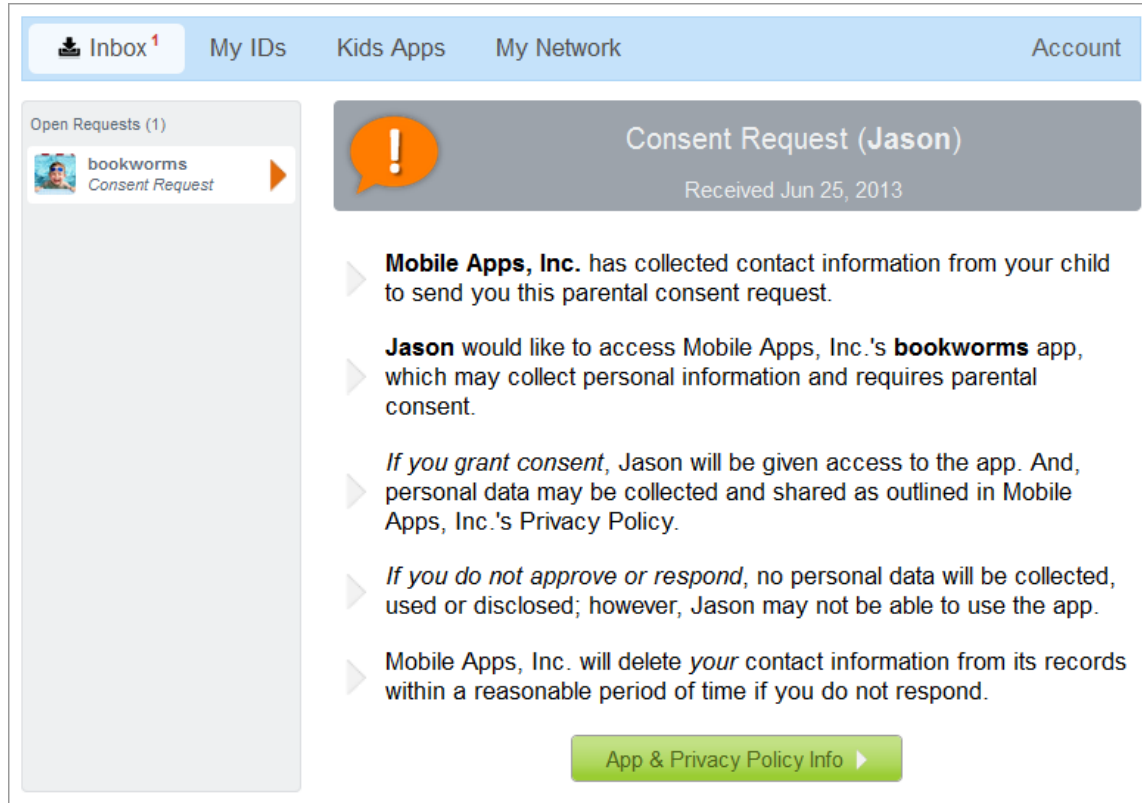
2.6.2.1 Direct Notices

All direct notices are presented within the Parent Portal. The representative screen-shot (below) shows screen-1 of a direct notice.

Screen-1 of the direct notice presents the parent with the following:

1. Name of the child for whom consent is requested,
2. Identity of the Operator requesting parental-consent,

3. Date of the request,
4. Name of the Application for which parental-consent is requested,
5. The nature of the request and the consequences of the parent's affirmative consent,
6. The consequences if the parent denies consent – no PII will be collected,
7. Notice that the parent's contact information will be deleted if no response is received.



Parent Portal - Direct Notice (screen 1)

The parent is directed to the application and policy information page after they have reviewed page-1 of the direct notice. There is only one process-flow through the direct notice – a parent is presented with all COPPA required information before they are presented with the means to respond to a request.

The representative screen-shot (below) shows screen-2 of the direct notice.

Screen-2 of the Direct Notice presents the parent with the following:

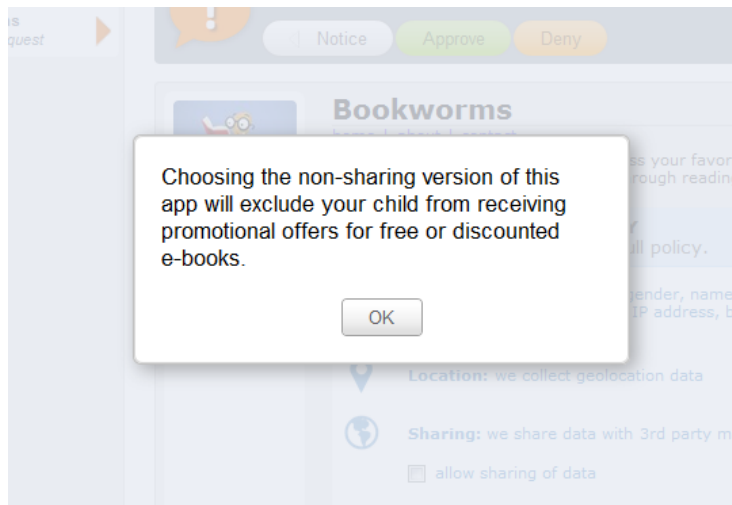
1. Name of the child for whom consent is being requested,
2. Name of the Application for which parental-consent is requested,
3. Brief description of the Application

4. Hyperlinks to the Operator's Home, About and Contact pages,
5. Hyperlink to the Operator's complete "General Privacy Policy",
6. Indication of the Application type (e.g. website, application, mobile-app, service , social-network),
7. Intended age-range for this Application,
8. Application Policy summary, including icons to communicate meta-data as well as details of the information collected, how it is used and how it is shared,
9. An (optional) Policy Brief describing why the Operator is requesting this information,
10. Approve and Deny buttons to capture the parent's request-response.

The screenshot displays a web interface for a 'Parent Portal'. At the top, there is a navigation bar with links: 'Inbox' (with a red notification badge), 'My IDs', 'Kids Apps', 'My Network', and 'Account'. Below this, on the left, is a sidebar titled 'Open Requests (1)' containing a card for 'bookworms Consent Request' with a small profile picture and a right-pointing arrow. The main content area is titled 'Consent Request (Jason)' and features a large orange exclamation mark icon, a 'Notice' button, and 'Approve' and 'Deny' buttons. Below the title, there is a section for the 'Bookworms' app, which includes a cartoon worm reading a book, the text 'MOBILE APP Ages: 4-14', and links for 'home | about | contact'. The app description reads: 'Where will knowledge take you? Discuss your favorite books with friends. There is a world to discover through reading.' To the right of the app info is a scrollable 'APP POLICY' section. It starts with 'Click [here](#) to view full policy.' and lists several data collection points with corresponding icons: 'Data' (person icon), 'Location' (location pin icon), 'Sharing' (globe icon), 'Social' (speech bubble icon), and 'Contact' (envelope icon). Each point has a brief description of what data is collected or shared. The 'Sharing' section includes a checkbox labeled 'allow sharing of data', which is currently checked and circled in red. The 'Social' section mentions that data sharing with peers is enabled on the social network.

Parent Portal - Direct Notice (screen 2)

For those Applications which share a child's PII with third-parties and for which the Operator supports a non-sharing version, a "**allow sharing of data**" option will be presented to the parent. Should the parent choose to decline this sharing option, they are presented with a pop-up displaying the Operator provided explanation of the differences (if any) between the non-sharing version and the sharing version of the Application.



Non-sharing Operator explanation

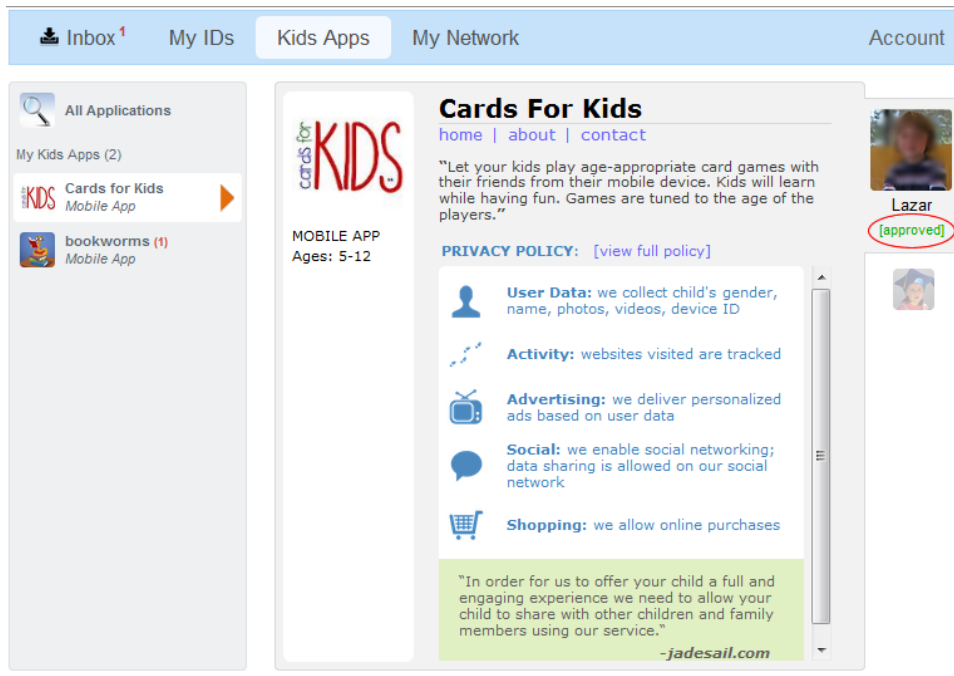
If a non-sharing version of the Application is not offered, the parent is informed that declining the sharing-option is equivalent to denying consent for this Application.

2.6.3 Consent Revocation (delete child's PII) Process

The Parent Portal allows a parent to access information for every consent they have granted through *ConsentID™*. By selecting the “Kids Apps” tab a parent can access these application specific details.

The left-hand column of the screen lists all Applications for which the parent has granted consent for one or more of their children. Selecting an Application in this column will display the details for that Application on the right-side of the screen.

In the representative screen-shot below, the “approved” status for the “Cards for Kids” application is highlighted. This screen indicates that the parent had previously granted consent for their child (Lazar) to access this application.



Parent Portal – Kids Apps tab (approved status highlighted)

Clicking on the “approved” status will present the parent with the pop-up shown in the following screen:



Parent Portal – Kids Apps tab (approval status displayed)

The pop-up displays the date consent had previously been granted for this child for this Application. At the parent's discretion, they may choose to revoke this consent (previously granted) or leave the consent unchanged.

2.6.3.1 Effects of Consent Revocation

Should a parent request that consent be revoked for an Application, this revocation is communicated back to the Operator via the *ConsentID™* API. Upon receipt of this revocation status, the Operator is required under the terms of the *ConsentID™* TOS to:

1. Discontinue the collection and use of the PII for this Child (for this Application)
2. Delete the Child's PII and disable the account for this Application

2.7 AssertID Verification Process

Central to the AssertID™ VPC method is AssertID's social-graph identity verification process. This process is employed to verify the identity of a parent as well as to verify each unique parent-child relationship. Some key benefits of this approach are:

1. AssertID's verification process does not require that a parent divulge sensitive personal or financial information such as SS#, address, bank account # or government ID and is therefore less intrusive to the parent.
2. Because less personal information is divulged the risk of unintended disclosure of personal information is diminished.
3. AssertID's process verifies the parent-child relationship using the same technology used to verify a parents' other identity attributes. This verification of the parent-child relationship distinguishes the AssertID VPC method from currently approved methods.
4. Upon achieving and maintaining a "passing" *trust score* (7 out of 10) for a given *ConsentID™*, no additional verifications are required. This is simpler for parents than currently approved methods and can result in quicker consent-responses.
5. Parents are able to pre-approve Applications for a child represented by an "enabled" *ConsentID™*. This can result in sub-second consent-response to a consent-request for this child for an Application that has been pre-approved.
6. Because no credit or debit card transactions are required, this verification method is non-discriminatory to individuals who do not have a debit or credit card.

2.7.1 Creating a User's AssertID

The first step in AssertID's social-verification process is for a user (in the case of *ConsentID™*, a parent) to create an AssertID. There are currently two use-cases for how this AssertID is created:

1. An individual (a parent) can visit the AssertID application and proactively create their own AssertID, or

2. When a parent receives their first *ConsentID™* consent-request from an Operator using the *ConsentID™* service, the parent is guided through the AssertID creation process.

2.7.2 Building a User's Social-graph

AssertID's social verification process begins by building a social-graph or "web of trust" from an individual's social network. This web of trust consists of friends and family members who know the individual sufficiently well to verify the individual's self-asserted identity attributes (e.g. name, location, age, gender, photo, children, etc.).

AssertID uses an individual's social-graph to:

1. Deliver verification-requests to the select individuals in the social-graph, and
2. To apply AssertID's proprietary technology to analyze this social-graph as part of the verification process

The method AssertID currently employs to create an individual's social-graph is to request that users authorize the AssertID application to access their Facebook profile. AssertID requests only "basic" authorization which is all that is required for AssertID to access the user's friends list.

From this friends list, the user builds their own social-graph by adding friends to their AssertID "My Verifiers" network. It is this "My Verifiers" network which constitutes the social-graph AssertID uses to verify the individual's self-asserted identity attributes. When a new person is added to the AssertID holder's social-graph, a personalized invitation is sent to that person asking them to confirm the identity attributes contained in the AssertID.

User's who either don't have a Facebook account or who prefer not to use the AssertID social-verification process are provided with the option to authorize consent-requests using the FTC approved credit-card method.

2.7.3 Trust-score calculation

AssertID's technology analyzes the nature and quality of all peer-verifications and from this analysis derives a quantitative measure (a "*trust score*") which is a reliable indicator of the likelihood that the identity attributes asserted by an individual are true.

AssertID calculates a *trust score* for each individual attribute, as well as an aggregate *trust score* for the individual represented by an AssertID. In addition, a separate *trust score* is calculated for each *ConsentID™*. This *ConsentID™ trust score* represents the verification of the unique parent-child relationship represented by the *ConsentID™*.

2.7.3.1 Verification Requests

Each individual added to the user's social-graph will be sent a personalized request asking them to verify the user's self-asserted identity attributes.

These "verifiers" are notified of these verification requests in two ways:

1. With a message within their Facebook account, and

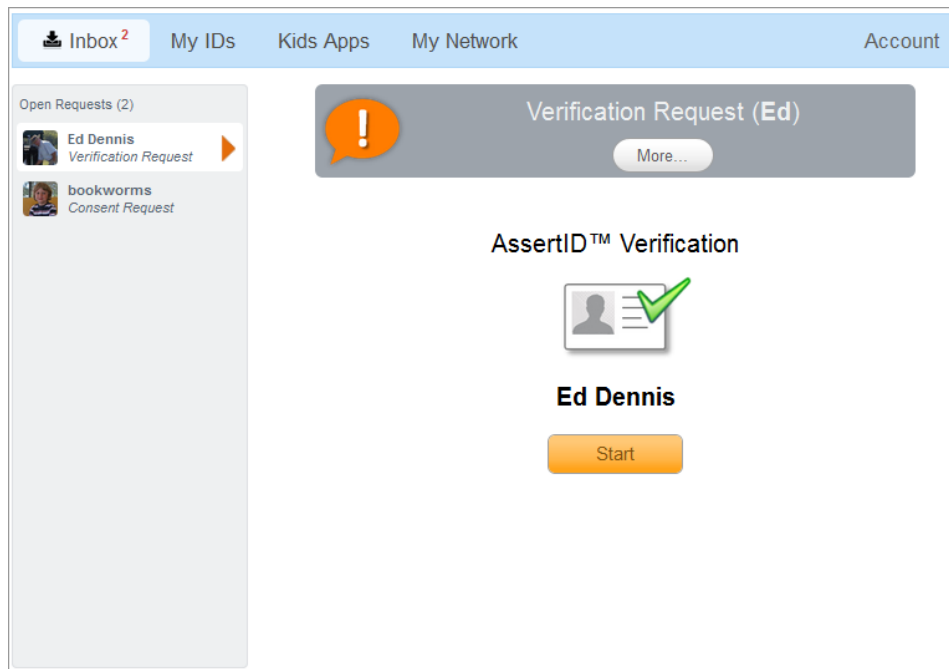
2. Verification requests are posted to each verifier's *ConsentID™* Inbox (if they have an AssertID).

2.7.4 Verification Process

During the verification process, the verifier is presented with each individual identity attribute contained on the AssertID of the individual (parent) requesting verification. The verifier is asked to provide one of three possible responses for each attribute presented:

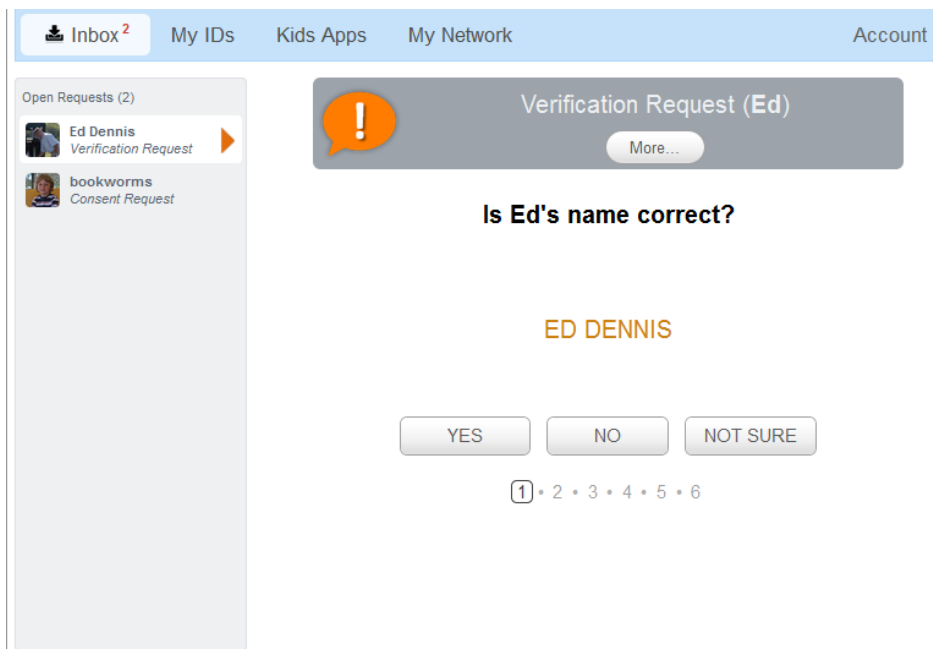
1. Yes – this attribute is accurate
2. No – this attribute is not accurate
3. Not Sure – cannot attest to the accuracy of this attribute

The representative screen-shot below shows how a verification-request is presented in the verifier's *ConsentID™* Inbox.



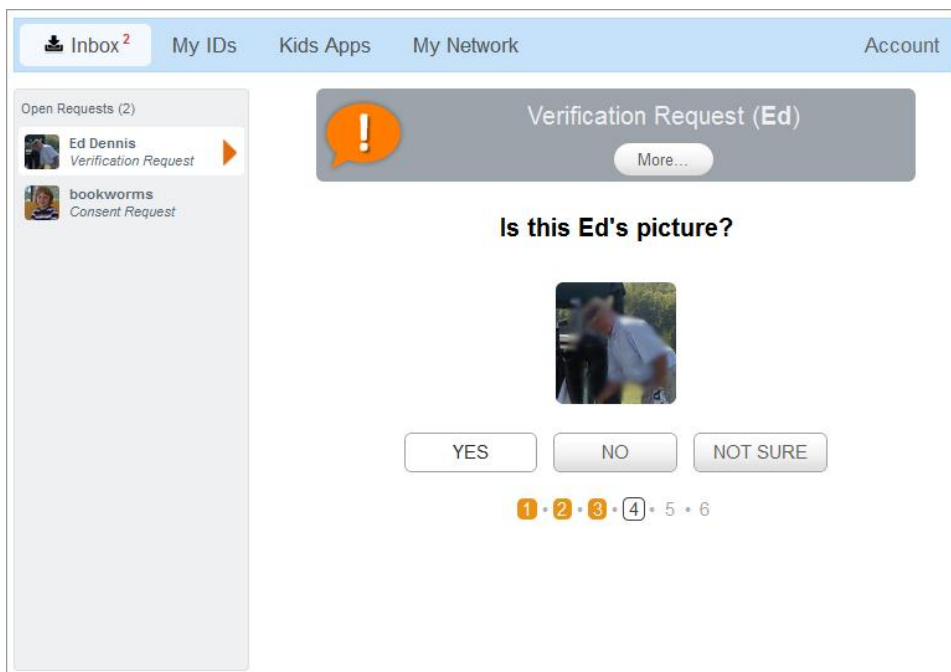
Verification-request start screen

Representative screen-shots requesting verification of the identity attributes of Name and Photo are provided below.



The screenshot shows a web interface with a top navigation bar containing 'Inbox²', 'My IDs', 'Kids Apps', 'My Network', and 'Account'. On the left, a sidebar titled 'Open Requests (2)' lists 'Ed Dennis Verification Request' and 'bookworms Consent Request'. The main content area is titled 'Verification Request (Ed)' with a 'More...' link. Below this, the question 'Is Ed's name correct?' is displayed. The name 'ED DENNIS' is shown in orange text. At the bottom, there are three buttons: 'YES', 'NO', and 'NOT SURE'. Below the buttons is a progress indicator: '1 • 2 • 3 • 4 • 5 • 6', where the number '1' is highlighted in a box.

Verification screen (Name attribute)



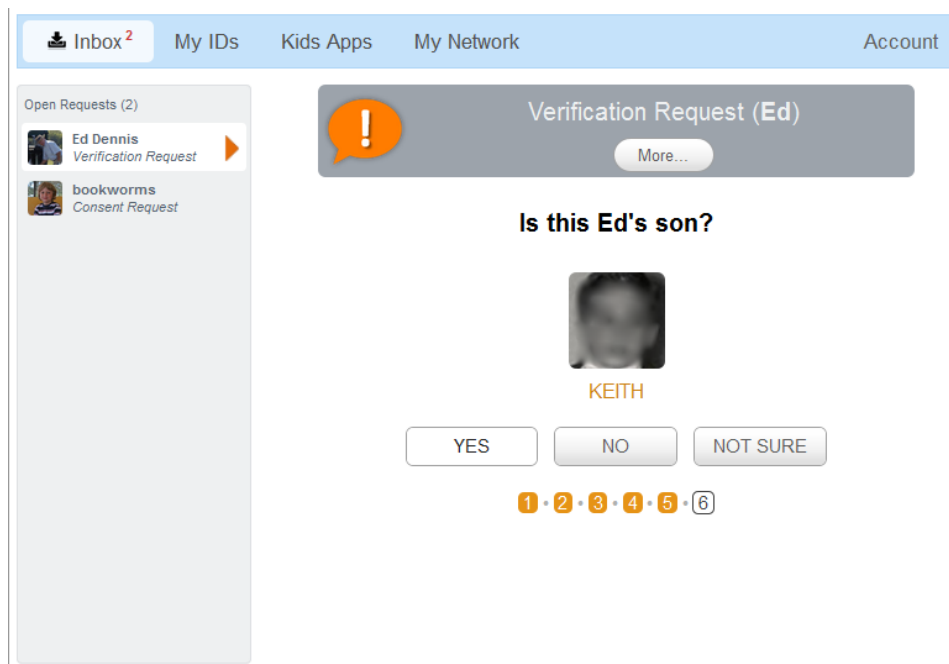
The screenshot shows a similar web interface. The top navigation bar and sidebar are identical. The main content area is titled 'Verification Request (Ed)' with a 'More...' link. Below this, the question 'Is this Ed's picture?' is displayed. A small photo of a man is shown. At the bottom, there are three buttons: 'YES', 'NO', and 'NOT SURE'. Below the buttons is a progress indicator: '1 • 2 • 3 • 4 • 5 • 6', where the numbers '1', '2', and '3' are highlighted in orange boxes, and the number '4' is highlighted in a box.

Verification screen (photo attribute)

Similar verification screens are presented for each attribute contained on the parent's AssertID. The numbers below the response buttons indicate the verifier's progress through the verification process.

In addition to the identity attributes that define the individual (parent), verifiers are also asked to verify each parent-child relationship.

The results of this parent-child relationship verification are reflected in the *trust score* of the associated *ConsentID™*.



Verification screen (Child attribute)

Should any “verified” attribute be changed by the user after verification (e.g. new photo), the *trust score* of that attribute is set to zero and all those who had previously verified the individual will be sent a verification request for the attribute that changed.

3 Analysis of AssertID’s VPC Method Compliance with Part 312.5(b)(1)

AssertID’s VPC method is designed specifically to satisfy Part 312.5(b)(1) COPPA Rule –

“An operator must make reasonable efforts to obtain verifiable parental consent, taking into consideration available technology. Any method to obtain verifiable parental consent must be reasonably calculated, in light of available technology, to ensure that the person providing consent is the child’s parent.”

The AssertID method uses available technology to achieve each of the following:

1. Verify the identity of the parent

2. Verify that the individual granting (or revoking) consent is the parent of the child for whom parental-consent was requested.
3. Restrict access to the consent-response mechanism to the parent(s) of the child.

3.1 Parent Identity Verification

To verify the parent's identity our method employs the AssertID's social-graph verification process as outlined in Section 2.7 above and as detailed in Exhibit A - "AssertID Verification Technology".

3.2 Parent-child Relationship Verification

AssertID employs two methods to verify the parent-child relationship:

1. AssertID's proprietary social-graph verification method (primary), and
2. The FTC approved credit-card verification method (alternative)

3.2.1 Social-graph Verification Method

The parent-child relationship is verified using the same process and technology as for verification of the parent's identity as outlined in Section 2.7 above, and as detailed in Exhibit A - "AssertID Verification Technology".

Each parent-child relationship is captured and represented by a unique *ConsentID*TM and its associated *trust score*. As the parent-child relationship represented by a *ConsentID*TM is verified by the parent's social-graph, the *trust score* changes to reflect these verifications.

Using this verification method, a parent can neither grant (nor deny) consent for the child represented in a *ConsentID*TM until the *trust score* achieved is 7/10 or higher thereby "enabling" the *ConsentID*TM. Provided a specific *ConsentID*TM is "enabled", a parent is able to respond to consent-requests for, or to proactively pre-approve Applications for the child associated with the *ConsentID*TM.

3.2.2 Credit-card method

An alternative (FTC approved) credit card verification method requiring a purchase transaction is offered for those parents who choose not to use the AssertID social-graph verification method or, who choose to provide immediate consent while waiting for their *ConsentID*TM *trust score* to reach 7. If the credit card verification method is used, a new purchase transaction is required for each consent granted.

AssertID's inclusion of this method as an alternative verification method is a purely practical consideration given that this method is FTC approved. This should not be interpreted to reflect AssertID's belief in the efficacy of this verification method.

3.2.3 Susceptibility to Fraud

No verification technology or method is 100% fool-proof; every verification method is susceptible to fraud. What is important is that the measures employed to prevent fraud – to prevent

someone from “gaming” a system - are appropriately robust given the value of the transactions being protected.

There is a natural tension or trade-off between the strength of these fraud-prevention measures and the ease-of-use of a system. The objective is to find a proper balance which provides acceptable fraud-prevention while at the same time does not introduce unnecessary friction into the process.

AssertID achieves this balance through our implementation of configurable “contexts” which allow us to adjust the veracity of our verifications to meet the specific needs of the application. In the case of the AssertID VPC method, we have tuned the weights applied to specific verification coefficients and variables within our proprietary *trust score* algorithm to achieve a balance between ease-of-use, veracity of verification and resistance to fraud.

3.3 Restricted Access to Consent Mechanism

All consent requests are delivered to the *ConsentID™* Parent Portal. To access this portal a parent must provide their *ConsentID™* login credentials, therefore only the parent of a child has access to the means to review and response to consent-requests for that child.

ConsentID™ separates the “Notification” that a consent-request is pending from the ability to access the consent-response mechanism (the Parent Portal) allowing greater flexibility in the methods used to deliver notifications.

This approach combined with the HTML5 implementation of the Parent Portal makes it possible for a mobile phone number to serve as an alternate means to notify the parent of a new consent-request allowing parents to respond to consent-request from their mobile phone.

The combination of items 3.1, 3.2 and 3.3 (above) provide a level of assurance that the individual providing consent is in fact the parent of the child for which consent is being requested that is significantly more rigorous than any currently approved method.

4 Exhibits

A. AssertID Verification Technology

Separate document attached

B. AssertID Patent Application “Method and System for On-line Identification Assertion” [\(AssertID Patent Application\)](#)



Promoting Trust and Civility in On-line Interactions

AssertID™ Verification Technology

Proprietary & Confidential

Contents

1	Introduction	3
2	AssertID™ Digital Identity	3
3	Digital Identity Context	4
4	Verification Process	4
5	Trust Score	5
6	Collusion	7
7	Applying Contexts	8
8	Summary	8

1 Introduction

AssertID™ is an online digital identity comprised of *self-asserted* attributes that are verified by one's network of friends and family.

Once the identity information is verified, the **AssertID™** can be used to process online transactions that require positive proof of identity (such as responding to parental consent requests).

The AssertID verification process is partly based on the theory of *social embeddedness* (Mark Granovetter), which suggests that the actions of individuals in a group are governed to some extent by their ties and social relations. More specifically, AssertID exploits the intrinsic *trust relationship* that exists between groups of friends and family. In these groups, there is no motivation or advantage to be gained for individuals to lie about each other's identity information. This provides an ideal environment for AssertID's verification process, which utilizes these strong social relations to validate one's identity information.

2 AssertID™ Digital Identity

An AssertID™ Digital Identity (**AID**) is a collection of self-asserted identity attributes that are tied to a real-world individual through an online account at assertid.com.

The process for creating and verifying an **AID** is as follows:

1. Create a password-protected account at assertid.com using a verifiable email address
2. Associate a social network to the assertid.com account (i.e. Facebook, etc.)
3. Select a list of friends and family from the social network that will serve as verifiers for the self-asserted attributes
4. Receive enough *positive* verifications from the verifiers to amass a *trust-score* of 7 (out of 10)
5. Maintain a trust-score of 7 to keep the **AID** enabled (i.e. if any attributes are changed after they have been verified, the verifications are reset and the attributes must be re-verified)

Once the above steps are completed, the **AID** is available as an online credential to facilitate transactions where positive proof of identity is required.

To use the **AID**, the individual selects it as his/her identity when performing a transaction in an online application that supports AssertID online identities. If the individual supplies the proper login credentials and his/her **AID** has a trust-score of 7 or above, the transaction is authorized and the individual's identity is confirmed.

This image consists of multiple horizontal black bars of different lengths stacked vertically. These bars represent redacted text or data points from a document. The bars vary significantly in length, with some spanning most of the width of the page and others being much shorter. There are approximately 20 such bars in total, grouped into several distinct sections separated by small gaps.

The validity of an AssertID identity is based on the successful verification of the individual's self-asserted attributes by a given number of verifiers from his/her social network. An **AID** is *valid for use* when the trust-score for that **AID** is 7 or above. An **AID** trust-score can vary from a minimum value of 0 to a maximum value of 10. The number of verifications required to achieve a trust-score of 7 for any given **AID** depends on the context in which it's being used as described above.

When verifying an attribute, verifiers are given the option of responding positively or negatively; or skip an attribute altogether if they are not sure. Positive verifications help the individual's trust-score. Negative verifications hurt the individual's trust-score. Skipped responses have no affect on the trust-score.

[REDACTED]

- [REDACTED]
- [REDACTED]
[REDACTED]
[REDACTED]
- [REDACTED]
[REDACTED]
- [REDACTED]
[REDACTED]
- [REDACTED]
[REDACTED]
- [REDACTED]
[REDACTED]
- [REDACTED]
[REDACTED]
- [REDACTED]
[REDACTED]

5 Trust Score

[REDACTED]
[REDACTED]
[REDACTED]

- [REDACTED]
- [REDACTED]
- [REDACTED]
[REDACTED]
- [REDACTED]
[REDACTED]
- [REDACTED]
[REDACTED]

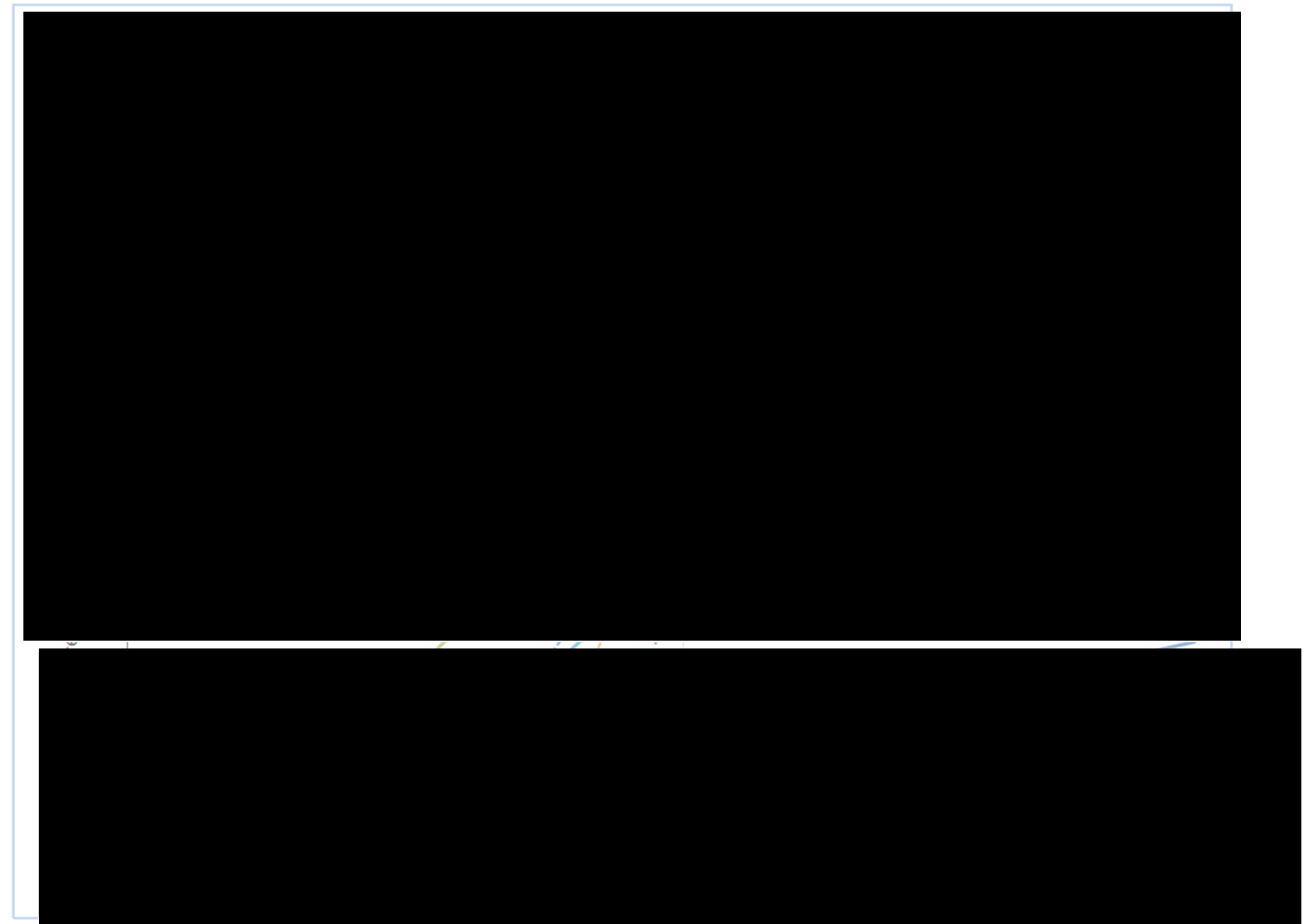
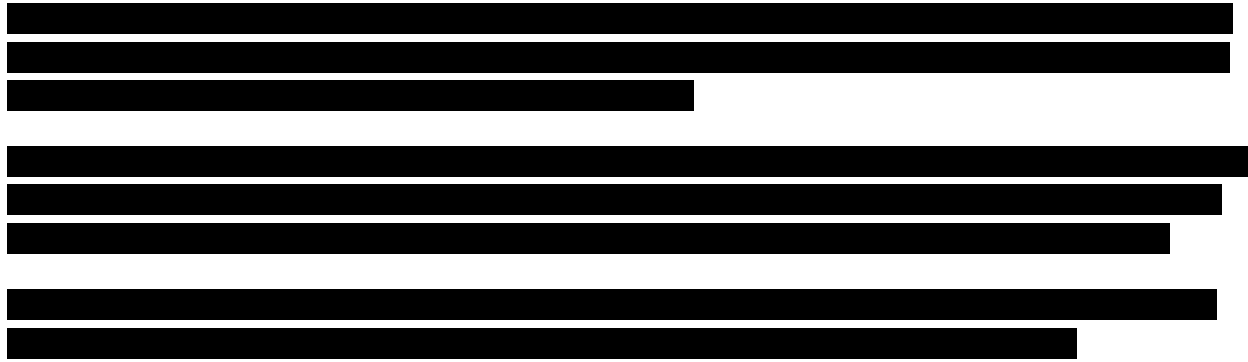


Figure 1: AID Trust-Score Calculation Curves



[REDACTED]

6 Collusion

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]



7 Applying Contexts



AssertID™ is the basic online digital identity that includes attributes such as name, age, gender, picture and location. This digital identity is currently only used privately by assertid.com to set up and verify a user's account. The trust-score of a user's **AssertID™** factors into the calculation of another user's trust-score when the holder of the **AssertID™** is acting as a verifier of other assertid.com members. In other words, the more trusted an assertid.com member is, the more he/she contributes to the verification of others.

ConsentID™ is an example of an AssertID digital identity applied to a specific context. In this case, the context is *verifiable parental consent*.

AssertID offers a parental consent service that enables operators wishing to comply with the FTC's COPPA Rule to ask for parental consent from members of assertid.com holding **ConsentID™** digital identities.

ConsentID™ is a digital identity that borrows some attributes of a member's **AssertID™** (i.e. name, age), but also incorporates parent-child relationship attributes that certify the member is the parent of a given child.

Parents that are holders of a verified **ConsentID™** can use it to respond to consent requests from operators needing to comply with the FTC's COPPA Rule. This COPPA Rule dictates that if the operator intends to collect personal information from a child under the age of 13, it must first get the parent's consent. The **ConsentID™** in this case serves to not only verify the parent's identity, but also the parent-child relationship.

8 Summary

AssertID's leading-edge verification technology transforms a *set of self-asserted (often public) identity attributes* into a verified Digital Identity that can be used to perform online transactions requiring definitive proof of identity.

AssertID's digital identities are easy to set up and use, employing a verification scheme that is in principle very straightforward; however, without compromising accuracy, reliability or security through the application of AssertID's sophisticated, state-of-the-art verification algorithm.

UNITED STATES PATENT APPLICATION

for

METHOD AND SYSTEM FOR ON-LINE IDENTIFICATION ASSERTION

Inventors:

Joon Nak Choi
PO Box 17467, Stanford, CA 94309

Kevin Trilli
1341 Bay Street, San Francisco, CA 94123

Correspondence via: Customer No. 26263

Attorney Docket No.: 12000044-0001-002

METHOD AND SYSTEM FOR ON-LINE IDENTIFICATION ASSERTION

RELATED APPLICATION

[0001] This application is a NONPROVISIONAL of and claims priority to U.S. Provisional Patent Application 61/035,330, filed March 10, 2008, incorporated herein by reference.

FIELD OF THE INVENTION

[0002] The present invention relates to methods and systems for verifying on-line identities and, more particularly, attributes of such identities (e.g., age, geographic location, etc.), using social network analysis and other means.

BACKGROUND

[0003] A. Introduction

[0004] “On the Internet, nobody knows you’re a dog.” This caption from Peter Steiner’s infamous cartoon, printed at page 61 of the July 5, 1993 issue of The New Yorker (Vol. 69, no. 20) and featuring two computer-savvy canines, embodies the essence of a serious problem in modern society. Although an ever-growing number of commercial and social transactions take place across electronic mediums, little if anything has been done to assist users of those mediums ensure that the other parties to those transactions are who they purport to be. That is, users of Web-based social networking sites, job hunting sites, dating sites, consumer-to-consumer commercial transactions sites, and a myriad of other, so-called Web 2.0 sites, have few, if any, means for verifying the identities or attributes of those they interact with in the on-line world.

[0005] Thus, the Web 2.0 revolution is built on an internal contradiction. The same technologies that have allowed companies to create borderless, virtual communities buzzing with social interaction and provide innovative and convenient ways for people to transact business, also prevent their users from knowing just who it is they are dealing with in those interactions. As a

result, newspapers and other media outlets report stories of sexual predators prowling social networks, preying on the young and innocent; bigots troll the forums, misleading and bullying community members; con artists haunt the marketplaces, defrauding on-line buyers and sellers; and members of on-line dating sites complain of dates who lie about their marital status, or look nothing like their posted photos. By enabling anonymous social interactions that foster creativity and connectivity, Web 2.0 enterprises unintentionally create opportunities for abuse at the same time.

[0006] B. Trust in Social Interactions

[0007] Whenever two people interact, they expect certain things from each other. Consider an example involving the purchase and sale of an article such as a laptop computer via an on-line commerce site. When the buyer and seller agree to the transaction, the buyer impliedly (or perhaps explicitly) promises to pay in a timely manner, and the seller (impliedly or explicitly) promises to send a product as advertised. In many cases, the buyer must believe the seller's promise (i.e., must trust the seller), and send payment before receiving the laptop computer. This involves a certain amount of risk: If the seller plans on abusing the buyer's trust, s/he could take the buyer's money without ever sending the laptop.

[0008] This example illustrates two important aspects of trust. Just like in the physical world, trust in the on-line world is often misplaced; not everyone honors promises. Second, trust creates the conditions for its own abuse; a person cannot be duped unless s/he trusts a scammer in the first place. Consequently, interactions present a social dilemma. For an interaction to occur, one of the two parties must act, trusting that the other party will honor her/his promises. Someone needs to make the first move.

[0009] For these reasons, people generally withhold trust unless they know something about another's trustworthiness. Most adults have an inner circle of trust: friends, family and close colleagues who have already proven trustworthy. They also tend to trust people who have been vouched for by a friend, or who have excellent reputations. In countries with strong legal systems, people will generally trust others to obey the law, at least in the absence of very strong incentives to break it. In contrast, reasonable adults typically distrust strangers in an off-line setting.

[0010] C. The Benefits of Radical Trust

[0011] Paradoxically, the same people who distrust real-life strangers often trust strangers in an on-line setting. They blog about intimate moments (revealing intimate details of their lives to anyone who cares to read about them), purchase items from unknown sellers (exposing themselves to fraud), and even swap homes with strangers. This is especially strange considering that face-to-face interactions provide far more signals about trustworthiness than on-line interactions. Body language, tones of voice and even the way someone is dressed all convey information relevant to questions of trust in the physical world. Some communication experts go as far as to suggest that 80% of face-to-face communication occurs through such non-verbal cues. Yet, people seem to trust on-line strangers more than offline ones. Why is this? Part of the answer lies in radical trust -- the belief that on-line community members should trust each other unconditionally.

[0012] Web 2.0 companies understand that they can build stronger communities -- and generate greater value -- by facilitating trust amongst community members. Many such companies live by O'Reilly's dictum: facilitate user interactions, and success will follow. Building community-wide trust is an important part of this process. Largely because they have fostered radical trust, Web 2.0 entities have grown tremendously.

[0013] D. The Dark Side of Radical Trust

[0014] However, radical trust has a dark side that is jeopardizing these achievements. Like any other form of trust, radical trust creates the conditions for its own abuse. If a community member ("Andy") trusts another ("Brad") to behave in a specified way, Brad can take advantage of Andy. Suppose that Andy is looking for a hotel room in a vacation spot, and so is reading reviews posted to an on-line travel advisory site before making a decision, and Brad is the proprietor of a motel in the area. Knowing that most readers of the on-line advisory site trust user reviews, Brad posts anonymous and misleading reviews of his run-down motel. Andy, trusting the community nature of the site, believes the review, visits Brad's motel, and ends up having a wholly unsatisfactory experience. Many users of on-line travel advisory sites complain about just such experiences and similar problems are found across several different kinds of Web 2.0 sites:

[0015] 1. User-generated content sites: Websites based on user-generated content (e.g., collaborative filtering sites, message boards, etc.) operate on an implicit assumption: content users can trust content providers to post accurate information. However, many people (like unscrupulous hotel proprietors) have an incentive to post misleading information. Notably, finance message boards are reputed to be flooded with false rumors and information intended to influence trading decisions that benefit the posters of the information.

[0016] 2. On-line dating sites: Like user-generated content sites, on-line dating sites depend on their users to provide accurate information. However, many on-line daters have incentives to embellish, omit or enhance important details (e.g., marital status or appearance). Thus, they post false information about themselves or photos taken when they were younger or in much better physical shape. Many on-line daters complain about such experiences. Additionally, dating sites need to be very careful not to allow anyone under the age of 18 into their sites to protect their users from potentially illegal contact with minors via their forums.

[0017] 3. Social networking sites: Social network businesses face a homologous problem; they depend on their users to post accurate profiles. Unlike the situation for on-line dating scenarios, not all profile misrepresentations have negative effects; users often post ridiculous ages (e.g., 99) or locations (e.g., Antarctica) as a joke. Yet, not all misrepresentations are harmless. Sexual predators often disguise themselves as children to gain their targets' confidence. Indeed, such practices are alarmingly widespread. A study by the National Center for Missing and Exploited Children found that 13% of all children using social network sites received unwanted sexual solicitations. Nearly a third of these solicitations were aggressive, meaning that the solicitor attempted to meet the child off-line. Additionally, 4% of children on-line were asked for nude pictures of themselves. ISAFE, a not-for-profit organization specializing in educating children on Internet safety, conducted a study that has shown the 1 in 5 children in grades 5-12 have met in person with someone they had originally met on-line. Additionally, with social network profiles and applications/widgets functioning much like business websites, spam is taking on a new form, sent by a supposed "friend" to an unknowing user.

[0018] 4. Commercial transaction sites: Auction sites and on-line marketplaces face a slightly different problem. Transactions are only possible if sellers trust buyers to pay, and buyers trust sellers to deliver. However, both sellers and buyers face strong incentives to cheat. Although

some on-line marketplaces have instituted countermeasures designed to punish cheaters, some types of abuse have nevertheless become commonplace, reducing the overall integrity of all such sites. For instance, shill bidding has pervaded on-line auction sites. In this practice, the seller (or someone in collusion therewith) registers fake bids on items for sale in order to prompt potential buyers into submitting higher bids. Also, high-reputation accounts (i.e., those which seemingly are associated with trustworthy individuals based on a marketplace reputation score) are available for purchase by fraudsters looking to make a quick sale of an expensive product to an unwitting buyer.

[0019] 5. Content providers. Radical trust can also extend to businesses interacting with consumers online. Providers of content intended for adult audiences (typically defined as Internet users older than 18 years old) have a challenging problem enforcing age restrictions for their sites due to this same inability to know who is accessing their sites. Typically, younger users with personal incentives to view this content game the system to appear to be an adult by simply using someone else's valid credit card. Perhaps worse, many sites simply ask users to self-assert their ages without undertaking any sort of validation.

[0020] E. Existing Solutions and their Inadequacies

[0021] Recognizing that radical trust can be abused, on-line businesses and web visionaries have proposed several solutions. Unfortunately, each of these "solutions" possesses exploitable weaknesses.

[0022] 1. Self-Regulation through Social Norms: Web 2.0 proponents propose that communities minimize abuse through self-regulation. In practice, self-regulation usually translates into a rhetorical exercise, where community leaders and the on-line business vigorously champion social norms ("community standards") against abusive behaviors. While such practices are easy and inexpensive to initiate and maintain, they tend to foster a false sense of security which creates opportunities for even greater abuse.

[0023] 2. Self-Regulation through Punitive Measures: A different kind of self-regulation involves punitive measures. A few on-line communities give their users the power to collectively rate each other. On many sites, bad ratings are linked with negative incentives. For instance, someone with a low rating on a commercial transaction site will have difficulty finding

transaction partners, who are scared off by a bad “reputation”. Thus, collective ratings systems give community members the power to punish repeat abusers. Nevertheless, while these measures have tended to reduce abuse, they possess known loopholes that are virtually impossible to adequately police. Moreover, site operators have almost no way to deter or prevent malicious users from perpetuating frauds with fresh accounts.

[0024] 3. Eliminating Web Anonymity: Compared with the off-line world, on-line communities offer an unprecedented amount of anonymity. To sign up for most on-line communities, users only need to present a valid e-mail address, available free from many different providers. Such addresses are virtually impossible to trace back to real-life individuals. As indicated above, for age verification most sites simply offer self-assertion, click-through agreements that push the age verification responsibility onto the user, without ever verifying that users' personal information.

[0025] Recognizing this problem, the South Korean government has outlawed on-line anonymity and now requires individuals to register their national identification numbers (equivalent to U.S. Social Security Numbers) with on-line communities they join. This requirement has reduced (but not eliminated) abusive practices. To eliminate abusive attacks altogether, the Korean government is implementing a “real names policy” where on-line community members will be identified by their real names, not on-line monikers. Already this “solution” has spawned other serious problems. Widespread usage of the national identification number has made it more vulnerable to theft, increasing identity theft across the country. More fundamentally, this requirement not only strips away the risks associated with Internet anonymity, but also its freedom-of-expression benefits. People are less inclined to voice unpopular opinions when they face physical-world retributions. Although Koreans were willing to give up this benefit, Americans are likely to place greater weight on these freedoms. Furthermore, a real-name policy conflicts with United States law, which prohibits the release of personal information about children under age 13. Thus, while a real-names policy may deter potential abusers from the most damaging trust abuses, it creates opportunities for widespread identity theft and is likely politically untenable in the United States.

[0026] 4. Reputation Systems: A more sophisticated version of a real-names policy links an individual's real name with his/her on-line reputation(s). Much like reputation mechanisms employed by on-line auction sites, emerging reputation systems ask users to rate their

interactions with one another. By providing such historical information, these companies attempt to address the Web 2.0 trust gap. Although groundbreaking in several ways, reputation systems face the same loopholes as less sophisticated ratings systems, and they lack any means for truly verifying the user-provided data (e.g., the user's real name) outside of crawling publicly available websites for confirmation, which must be assumed to provide only self-asserted, untrusted data. Thus, despite these efforts, users of these on-line services remain, for all practical purposes, anonymous.

[0027] This anonymity exposes a fundamental flaw in the reputation system model -- community members with "bad" reputations can always start over with a new profile. Even worse, nothing stops a user from creating dozens of profiles (each under a different user name, for example), and using them to falsely enhance a fake profile's reputation through positive reviews. Just as importantly, users who register legitimate complaints face retaliation from their abusers.

[0028] Additionally, reputation system ratings are difficult to interpret. Unlike on-line auction site ratings, which cover interactions occurring in a well-defined marketplace, reputation systems generally attempt to create a unified reputation spanning multiple social spheres. Unfortunately, a user's reputation in one sphere may not be relevant in another. Often, reputations are subjective and require a great deal of interpretation. Thus, reputation ratings have the potential for creating more confusion than they alleviate and while they may reduce some information shortfalls (because individuals may act to protect their reputations), it remains virtually impossible to deter malicious users from starting over with a fresh account.

[0029] 5. MySpaceTM: MySpace has become one of the most popular social network sites for minors and faces particular problems in protecting these children against predation by child molesters. To combat this threat, MySpace has made all 14- and 15-year old members' profiles private, making them accessible only to the adolescent's immediate friends. Additionally, MySpace is trying to keep younger adolescents from being contacted by adult strangers. While admirable, this initiative is fundamentally flawed. On one hand, nothing stops a potential abuser from lying about his/her age in his/her profile. On the other, adolescents often claim that they are 18 or older, often as a direct reaction against restrictions that are intended to protect them from potential predators. Without a means of verifying self-reported information, the MySpace initiative cannot succeed.

[0030] 6. PGP's Web-of-Trust: An alternative model is based on physical-world notions of trust between individuals. Most people have an inner circle of trust, composed of friends, family and close colleagues. Such people might not trust strangers, unless a trusted confidante vouched for them. For instance, consider three people, Adam, Benjamin and Carol. Suppose Adam does not know anything about Carol, but trusts his close friend Benjamin, who in turn knows and trusts Carol. In this situation, Benjamin could introduce Carol to Adam as a trustworthy person. Using this principle, the PGP (Pretty Good Privacy) Web-of-Trust extends a network of trustworthy people to the on-line world. An individual can be connected with a stranger through a chain of trust, where each link represents a person vouching for another. This system can conceivably be adapted for wider usage within Internet communities. If an on-line system were to track people who vouched for each other, the members of this network could constitute an enlarged circle of trust. These people could even remain anonymous to each other.

[0031] Although intriguing, this concept is not as robust as it appears. The PGP Web-of-Trust connects two people using a single chain of individuals who vouch for each other. Consider then a situation where a single person in that chain misplaces his/her trust, mistakenly (or intentionally) vouching for someone who is not trustworthy. The untrustworthy individual can then vouch for other untrustworthy individuals, and the entire system collapses. Thus, the PGP Web-of-Trust could potentially be brought down by a single point of failure. Further, the method of vetting new members in a web of trust is handled in a one-on-one, in-person inspection of government-issued identity documents. This process is very difficult to scale beyond a few users and rollout in a global on-line community. Thus, while the Web-of-Trust leverages physical-world manifestations of interpersonal relationships and trust, it possesses no redundancy mechanisms leaving it vulnerable to a single point of failure (a breach of trust) that can collapse the overall system's integrity.

SUMMARY OF THE INVENTION

[0032] The present invention provides methods and systems for verifying self-asserted socio-demographic attributes of individuals' identities, using social network analysis and other means. Through these processes, parties to a transaction or interaction are provided a measure of confidence about another party's self-asserted socio-demographic attributes, such as age, gender, marital status, etc., in order to assist in determining whether or not to pursue the transaction or interaction. The measure of confidence may be provided as a quantitative "score" indicative of the likelihood the user's self-asserted attribute is actually true. The quantitative score is derived by analyzing a web of trust in which the user is embedded.

[0033] In one embodiment of the invention, a quantitative measure of a trustworthiness of a self-asserted attribute of an individual is determined through a combination of analysis of a social network of which the individual is a member and non-network based analyses, and reporting said measure.

[0034] In a further embodiment of the invention, a credential is reported in response to receipt of a request therefor. The credential represents an estimate as to how likely a self-asserted attribute of an individual representing said attribute as true is in fact true. The estimate is computed through a plurality of mechanisms, including an examination of a web of trust within which the individual is embedded and non-network analysis based measures of the veracity of the attribute's asserted value.

[0035] The examination of the web of trust may include computing a contribution for embeddedness of the individual in the web of trust, for example computing contributions for direct and indirect embeddedness of the individual in the web of trust. The non-network analysis based measures may include identity measures which reward the individual for association with user profiles including difficult to replicable elements, and verification of the attribute with information obtained from trusted sources outside of the web of trust.

[0036] In some embodiments of the invention, the estimate is computed using weighted contributions for direct embeddedness of the individual in the web of trust, indirect embeddedness of the individual in the web of trust, embeddedness of the individual social in networks other than the web of trust, identity measures which reward the individual for

association with user profiles including difficult to replicable elements, and verification of the attribute with information obtained from trusted sources outside of the web of trust. In some cases, contributions for direct embeddedness of the individual in the web of trust are determined according to a computation of the individual's centrality within the web of trust (e.g., using a modified version of indegree Bonacich centrality). Contributions for indirect embeddedness of the individual in the web of trust may likewise be determined according to a computation of the individual's centrality within the web of trust this time using a different modified version of indegree Bonacich centrality, including one modification limiting a total indirect embeddedness contribution per verifying member of the web of trust for the individual. The contributions for indirect embeddedness may be capped at a threshold so as to guard against undue contributions for redundant verification paths, etc.

[0037] In some instances the estimate is computed through a scoresheet approach in which the individual mechanisms by which trustworthiness of the self-asserted attribute is measured are each allocated a specific number of scoresheet points and a credential's score is a summed total of the scoresheet points. Contributions to the credential score for indirect embeddedness of the individual in the web of trust may make up a majority of the scoresheet points for the examination of a web of trust within which the individual is embedded. Contributions to the credential score attributable to verification of the attribute with information obtained from trusted sources outside of the web of trust may make up a single largest component of the scoresheet points.

[0038] A further embodiment of the present invention involves quantitatively measuring an individual's embeddedness within a social network and assigning a score thereto, combining that score with a quantitative measure of the veracity of the attribute's asserted value as determined through non-network based analysis to produce a combined score, and reporting the combined score as a measure of trustworthiness of a self-asserted attribute of the individual. In such cases, measuring the individual's embeddedness within the social network may include determining contributions for the individual's direct embeddedness and indirect embeddedness in the social network. As indicated above, a contribution for the individual's direct and indirect embeddedness in the social network may be determined by computing the individual's centrality within the social network. The non-network analysis may include a quantitative contribution for identity measures indicative of the individual's association with user profiles including difficult

to replicable elements and/or verification of the attribute with information obtained from trusted sources outside of the social network. The combined score may be computed through the scoresheet approach in which each quantitative measure is allocated a contribution to the combined score up to a threshold.

[0039] These and other features of the present invention are described in detail below.

BRIEF DESCRIPTION OF THE DRAWINGS

[0040] The present invention is illustrated by way of example, and not limitation, in the figures of the accompanying drawings, in which:

[0041] Figure 1 illustrates relationships between participants of an interaction/transaction in the context of the present invention.

[0042] Figure 2 illustrates varying relationships between credential holders, direct verifiers of the credential holders and indirect verifiers of the credential holders for two different network cases.

[0043] Figure 3 illustrates differences in network relationships between a closely-knit group of individuals and a loosely knit group of individuals.

[0044] Figure 4 illustrates differences in indegree Bonacich centrality between networks exhibiting significant closure and those exhibiting reduced degrees of closure.

DETAILED DESCRIPTION

[0045] Described herein are methods and systems for verifying on-line identities and, more particularly, attributes of such identities, using social network analysis and other means. As used herein, the term “identity” is meant to encompass individual characteristics by which a thing or person is recognized or known. In one embodiment, these methods and systems are implemented so as to provide a measure of confidence about a user’s self-asserted socio-demographic attributes, such as age, gender, marital status, etc., and make that measure available to others seeking to determine whether or not a user is who the user purports to be or possess attributes he/she purports to possess. The measure of confidence may be provided as a quantitative “score” indicative of the likelihood the user’s self-asserted attribute is actually true. As used herein, the term likelihood is not intended to convey a probability but rather a measure defined by the algorithm discussed below. The quantitative score is derived in two stages: (1) building a web of trust amongst users of the service, and (2) computing those users’ embeddedness within the web of trust.

[0046] Embodiments of the present invention may take the form of an on-line service having a front-end functioning as identity oracle, collecting and warehousing private information about on-line individuals, and a back-end that functions as a web-of-trust verifying self-asserted information about its users. The information so collected and verified can be made available (either in raw form or, preferably, in the form of or accompanied by the qualitative score) to answer questions or provide assurances about an individual’s self-asserted attributes -- in some cases without actually disclosing the private data. Consider a hypothetical example. A user (ID:123) applies to join Club Penguin, an on-line social network open only to minors. To determine whether or not 123 is really a minor, Club Penguin queries the identity oracle about 123’s age. Because the identity oracle possesses private information about 123 (e.g., that he is John Doe, age 12, living at 123 Main Street in Anytown), the identity oracle is able to verify 123’s age (either by releasing same to Club Penguin or simply by answer the query affirmatively) while keeping 123’s other attributes private.

[0047] Among the features that set the present invention apart from solutions such as those discussed above is verification of users’ self-asserted attributes. Most on-line communities today trust their users to tell the truth about themselves -- i.e., to self-assert accurate data about

themselves. Yet, many users self-assert false information. For instance, sexual predators sometimes pretend to be minors to gain their intended victims' confidence. To limit such misrepresentations, the present invention uses the following logic:

1. In the absence of age verification, any user can lie about his or her age (or other attribute). Thus, users' self-asserted ages (or other attributes) cannot be assumed as accurate.
2. A typical user is connected with people on-line who know him/her off-line -- physical-world friends and colleagues. These people know something about the user's real age (or other subject attributes).
3. If such people verify that the user is telling the truth about his/her age/attribute (vouching for the user), even outsiders (i.e., strangers) can have greater confidence in the user's self-asserted age/attribute.
4. Users verified by many other users can be trusted more than users verified by few other users.
5. Users verified by other users who themselves have been verified can be trusted to an even greater extent; they are verified by others known to be trustworthy.

[0048] As indicated, age is only one of several socio-demographic attributes verifiable through this logic. Gender, marital status and geographic location can be verified in much the same way. The present invention provides an easy-to-interpret score representing the likelihood that a user is self-asserting his actual age or other attribute. These scores are computed by an algorithm based on social network analysis. Thus, the present invention enhances the identity oracle concept, by providing not only users' self-asserted ages, but also its degree of confidence in this data.

[0049] The same approach has many applications. For example, it can limit/prevent minors from accessing inappropriate web content. When an on-line user applies to enter an adult-only website, the site may query the identity oracle about the user's age. If the identity oracle is reasonably sure that the user is 18 or over (21 in some jurisdictions), the site grants user access. The identity oracle can also reduce online harassment and bullying using a similar approach.

Cyber-bullies gain much of their power by misrepresenting themselves online. If online communities validate users' self-asserted attributes (e.g., age, gender, etc.) using the facilities of the identity oracle, bullies will find it much more difficult to misrepresent themselves. Thus, the present system provides its users information about each other, empowering them to make more accurate trust judgments (i.e., judgments concerning each other's trustworthiness).

[0050] Before proceeding, it is useful to precisely define the problem space within which the present invention finds application and create a concise vocabulary for concepts used throughout the remainder of this discussion. As we observed above, in a typical interaction or transaction one user must take a leap of faith: making himself vulnerable to the other user (for instance, by pre-paying for an as-yet-undelivered laptop computer). When neither user trusts his/her counterparty enough to make this leap of faith, interactions/transactions fail to take place. Conversely, trust abuses occur when one user takes the leap of faith, and the counterparty.

[0051] The present systems and methods alleviate these problems by providing a more accurate basis for trust judgments about its users. Users can make more accurate trust judgments when they have reliable information about each other's socio-demographic attributes. This has two consequences. On one hand, parties to a transaction have more confidence in each other's trustworthiness. Interactions become less risky in general, and, consequently, become more frequent. On the other hand, spoofers (e.g., those intending to not honor promises and/or deceive other users) have less ability to hide behind Internet anonymity. Users can avoid spoofers more easily, decreasing opportunities for misplaced trust.

[0052] Referring now to Figure 1, we introduce the participants of an interaction/transaction and their relationship to one another. In the present discussion, the user who takes a leap of faith in a transaction or interaction 10 is labeled a relying party (RP) 12 because s/he relies on the present system to provide accurate information about a counterparty. This RP receives a system-issued credential 22 indicating a confidence that the other party to the transaction or interaction, a credential holder (CH) 14, is not self-asserting false socio-demographic attributes. Each RP may him/herself be a CH.

[0053] In this context, the "system" may, in one embodiment, be an identity oracle fashioned in the manner described above. More generally, such a "system" may be an on-line (e.g., Web-based) service configured to provide verified, self-asserted information about its users or

confidence scores indicative of a level of certainty or confidence that certain user-asserted attributes are true. By Web-based, we mean a service hosted at a computer-based resource accessible via a computer network (or network of networks) through convention tools such as an Internet browser.

[0054] For any given user, the system generates a credential by examining how that user is embedded in the system's web-of-trust 20: a social network of registered users who have verified each other's attributes. In other words, the system generates a credential for a specific attribute self-asserted by the subject CH by examining which other users validate that attribute (i.e., vouch for the CH's veracity). Such users are called direct validators (DV) 16₁, 16₂.

[0055] A user may be a DV in the context of one interaction, but be a CH in another interaction. Thus, DVs may have received validations of their own. In the context of the original interaction, the users who have validated DV attributes become validators-of-validators for the CH. Such users are labeled indirect validators (IV) 18.

[0056] As will become more apparent from the discussion below, in the present methods and systems users do not directly assess each other's trustworthiness, but end up doing so indirectly, to the extent that they trust each other to self-assert true identity attributes. Consider user A, who validates another user B's attributes. By doing so, A is indicating his belief that B is telling the truth. This says something about B's trustworthiness as a user. Thus, attribute validations serve as a proxy for user validations. Moreover, as users validate each other's attributes, they build a network of implicit user-level validations. Users who are more "entangled" in this network can be trusted more than their less-entangled peers because they have been verified by many users -- who themselves have been verified by still other users. This builds on sociological research finding that: (1) human beings are "embedded" (i.e., entangled) in webs of social relationships; (2) the way they are entangled (i.e., embedded) affects their behaviors; and (3) with greater embeddedness in a social network, people are less likely deceive and/or cheat other members of that network. The final point speaks to an important consideration: greater embeddedness indicates greater trustworthiness. This is explored further below.

[0057] A. Differentiating Between User- and Attribute-Validations

[0058] In the context of a specified interaction, DVs validate CH attributes (24), while IVs validate DVs as users (26). On one hand, users do not validate other users, but rather validate their attributes. A CH self-asserts many different attributes. A given DV may know about one CH attribute, but lack information about others. Thus, that DV may validate some of the CH's attributes, but not others. Thus, DVs validate attributes, not the CH as a whole user.

[0059] The most salient aspect of social relationships is trust; nearly all social network analyses implicitly analyze trust between individuals. The present system uses attribute validations as a proxy for trust between its users. Although trust constitutes the core of a social relationship, academic analysts seldom analyze trust relationships themselves; it is nearly impossible to collect data on trust itself. Data (users validating other users' attributes) translates to interpersonal trust in a straightforward manner. If user A validates another user B, A can be assumed to trust B to the extent that A believes that B is self-asserting true attributes. Thus, A's validation of B's attributes says something about A's trust in B as an individual. Accordingly, the present system uses attribute-level validations as a useful proxy for user-level trust relationships, a major component in its analyses.

[0060] The strength of these inter-user relationships is a closely related issue. Some relationships are stronger than others; for instance, friendships are generally stronger than acquaintances. The number of attribute-level validations can, therefore, represent a straightforward proxy. The more information two people know about each other, and the greater the number attributes they are willing to verify about each other, the more likely that they share greater trust. For instance, consider that if a user A validates six of another user B's attributes, the A-B relationship is likely stronger than another relationship between users C and D, where C validates only four of D's attributes.

[0061] Trust is dichotomized at a "strong acquaintance" level (people who know each other and have spent a little time together, but are not necessarily friends). This level is meaningful because it includes everyone who really knows the person, while at the same time excluding others who may have met the person a few times yet lack a meaningful social relationship. Thus, this threshold captures everyone in a network who has reliable data about an individual, but excludes others who have incomplete or potentially incorrect information. For these reasons, in one embodiment of the present invention one user (A) will be considered to have validated

another (B) if A has validated a complete “basket” of B’s basic attributes. This basic basket of attributes may include attributes that tend to include information known among people that share a meaningful relationship, for example a user's name, address, gender and birth date (age). Stated differently, the basket of attributes may include only those attributes that anyone who knows a user in any meaningful way should know. Other attributes (e.g., a current job, a place of birth, etc.) are excluded because it is possible to know people in a substantially meaningful way without knowing these attributes. Once A has validated every one of B’s attributes in the basic basket of attributes, A can validate more esoteric attributes.

[0062] These explanations provide a basis for answering the question posed above: why DVs validate CH attributes while IVs validate DVs (as users). DVs know the CH directly, and have a basis for personally validating the CH’s attributes. In contrast, IVs have no such relationship with the CH (by definition). Thus, the only way they contribute towards assessing the CH’s trustworthiness is by (1) validating DVs; (2) making them more trustworthy (i.e., raising their SU scores (see below)); and (3) allowing them to validate the CH’s attributes with greater weight. Thus, IV validations are necessarily filtered through the IV-DV relationship.

[0063] The present system differentiates attribute- and user-level validations. It issues an attribute score (SA) as a credential indicating a degree of confidence in a subject CH attribute. SA is returned to all users who query the system regarding the relevant CH attribute. In contrast, the system uses the user score (SU) when computing SA (discussed below). In one embodiment of the invention, for the attributes in the basic basket, $SA = SU$, because all basic attributes receive the verifications from the same people.

[0064] As illustrated diagrammatically in Figure 1, when the RP queries the system about CH’s attributes, the system returns its estimate how likely these attributes are to be true. To compute this estimate, the system examines the web of trust that the CH is embedded within, in addition to non-network signals of attribute truthfulness.

[0065] More precisely, the system will measure CH’s embeddedness in the system’s web of trust (along with other relevant signals), and return the results to the RP, packaged as a credential. In one embodiment of the invention, a social network analysis (SNA) -based algorithm is used to generate an accurate quantification of identity trust from the network of self-asserted attributes.

This following discussion introduces the principles behind such a process and builds an example of the process from these principles.

[0066] B. Embeddedness and Egocentrism

[0067] The system measures a CH's embeddedness in social networks. This concept refers to human beings' "entanglement" in webs of ongoing social relationships. In general, human beings are entangled in webs of social relationships (i.e., social networks); the way they are entangled (i.e., embedded/dis-embedded) affects their behaviors; and with greater embeddedness in a social network, people are less likely to deceive and/or cheat other members of that network. Since greater embeddedness indicates greater trustworthiness, the present system quantitatively measures a CH's degree of embeddedness.

[0068] Compared with their less-embedded peers, highly-embedded CHs (ones that are more difficult to dis-embed) have two characteristics: (1) They are verified (trusted) by a greater number of DVs; (2) who in turn are each verified (trusted) by a greater number of IVs. A CH's degree of embeddedness can be related to the CH's centrality. Measures for centrality come in several different varieties, each oriented to different objectives.

[0069] Local centrality measures a user's embeddedness within the individual's local network (radiating out from the individual), while global centrality measures a user's embeddedness within a network as a whole. For purposes of the present invention, local centrality measures are more relevant than global centrality measures, primarily because trust degrades quickly over social distance. For instance, most people trust their friends, and tend to trust friends-of-friends. However, they tend not to trust friends-of-friends-of-friends -- people who are so distant that they are practically strangers. Thus, socially-distant people do not contribute much towards a CH's trustworthiness. In other words, it is a CH's embeddedness in a local web of trust that really matters, not his/her embeddedness in the larger web. This is also consistent with a usability requirement of a system such as that being presently proposed: The goal is to obtain a certain level of usable trust without imposing significant friction on the user, as security systems are not usually the primary goal of a user, they are, however, necessary to permit safe interactions in a social or entertainment network.

[0070] Local centrality measures come in two varieties. Degree centrality counts the number of individuals who are connected to the focal individual; similarly, indegree centrality counts the number of individuals who are “pointing at” the focal individual. Bonacich centrality layers greater sophistication on top of degree centrality.

[0071] In particular, Bonacich centrality is a function of a focal individual’s number of connections, with each connection weighted by the value of its connections. In other words, a focal individual gains greater Bonacich centrality by connecting with well-connected vs. relatively isolated individuals. In mathematical terms, the local centrality of node i in a social network (graph) with j connections is calculated by:

$$C_i = \sum_j r_{ij} (\alpha + \beta C_j)$$

where C_i is the centrality of node i , r_{ij} is the (quantified) strength of the connection between individuals i and j , and C_j is the centrality of node j . α is an arbitrary standardizing constant ensuring that the final centrality scores will vary around a mean value of 1. In contrast, β has more substantial significance; it indicates how much C_j , should contribute towards C_i . $\beta = 1$ indicates that the full value of C_j is added to C_i ; in contrast, $\beta = 0$ indicates that the C_j does not affect C_i at all. Where r_{ij} and α both = 1 and $\beta = 0$, the equation for Bonacich centrality reduces to the equation for (un-normalized) degree centrality.

[0072] In various embodiments of the present system, degree centrality becomes the size of a focal individual’s immediate social circle. It shows how large the CH’s immediate circle of trust is, and therefore, how trustworthy the CH is likely to be. Indegree centrality is even more useful; it becomes a count of users (DVs) that validate (“point towards”) the CH. These measures usefully illustrate the CH’s embeddedness in an immediately local network.

[0073] According to another embodiment of the invention, a version of Bonacich centrality that counts only incoming connections (indegree Bonacich centrality) may be used. This measure starts with indegree centrality, but radiates out further in the network. To understand this, consider the example of two different networks shown in Figure 2. Here two CHs (CH_A 24 in network 1 and CH_B 28 in network 2) each receive a single DV validation. However, CH_B is validated by a DV 30 receiving an IV 32 verification, while CH_A is validated by a DV 26 that lacks any IV validation. Here, CH_B is more embedded in his local network than CH_A is in her

local network. Unlike indegree degree centrality, indegree Bonacich centrality accounts for such differences. r_{ij} is constant for all verifications.

[0074] To better understand the above, consider that if $r_{ij} = 1$, $\alpha = 1$ (not standardized), and $\beta = 0.5$, then $C_A = 1$ and $C_B = 1.5$, as follows:

$$\begin{aligned}\text{For CH}_A: C_A &= r_{ij}(\alpha + \beta * C_{DV26}) \\ &= (1)(1 + 0.5 * 0); C_{DV26} = 0 \text{ because DV 26 is not verified by any IV.} \\ &= 1\end{aligned}$$

$$\begin{aligned}\text{For CH}_B: C_B &= r_{ij}(\alpha + \beta * C_{DV30}) \\ &= (1)(1 + 0.5 * 1); C_{DV30} = r_{ij}(\alpha + \beta * C_{IV32}) = (1)(1 + 0.5 * 0) = 1 \\ &= 1.5\end{aligned}$$

There are no cycles (i.e., loops in the social network graph), so centrality scores for each network are computed in a single iteration.

[0075] Indegree Bonacich centrality not only measures a CH's entanglement in his/her immediately local (DV) and slightly-removed (IV) networks, but also matches intuitively with the butterfly-in-a-web metaphor (it takes fewer cuts to remove a butterfly entangled in a remote part of a spider's web than it does to remove a butterfly entangled near the center of a web). Consequently, indegree Bonacich centrality represents a substantively meaningful measure of a CH's embeddedness into his/her local network; it appears to be a reasonable measure of embeddedness.

[0076] C. Relational Non-Redundancy

[0077] However, Bonacich centrality is not a perfect solution. For example, this measure does not account for the way redundancy affects trustworthiness. Redundancy, in this context, refers to the existence of multiple chains of relationships (paths) connecting two individuals in a social network. Individuals who are connected with a greater number of unique (non-overlapping) paths are more difficult to disconnect from each other. For instance, consider two individuals connected through seven unique paths. To cut information flows between these individuals, one would have to sever seven distinct communication channels. In contrast, two individuals

connected through a single unique path could be disconnected by severing that one communication path.

[0078] A social network can be considered “more redundant” if it contains a higher proportion of redundant paths compared to a “less redundant” network. Egocentric social networks range between two extremes: complete redundancy (where everyone is connected with each other) versus complete non-redundancy (where no redundant paths exist). People face a trade-off between these extremes. Why? At any given time, a person can only maintain a finite number of social relationships; each relationship takes time to maintain, and people have a finite amount of time. Given this situation, a person has choices ranging between the two extremes: to maintain relationships with a closely-knit group of friends who all know each other -- illustrated diagrammatically as CH_C 34 of network 3 in Figure 3, -- or to share relationships with a widely dispersed group of individuals that do not know each other -- illustrated diagrammatically as CH_D 42 of network 4.

[0079] Which network, 3 or 4, is more advantageous? The answer depends on the situation. For many purposes (e.g., building a community), network 3 is more advantageous. However, for the purpose of obtaining unique information (i.e., networking to find a job), network 4 is advantageous. Individuals who do not know each other more likely obtain information from different sources, and the information they provide is more likely to be diverse. In contrast, information that originates within a close-knit group of people is likely to spread quickly within that network, crowding out other relevant pieces of information. Thus, the focal individual CH_C 34 of network 3 is likely to receive the same (redundant) information from many different people; for instance, s/he might find out about the same job opening from several of his friends (who all know each other). In contrast, CH_D 42 of network 4 is likely to receive different (non-redundant) information from many different people; for instance, s/he might learn about several different job openings.

[0080] For the situations depicted in Figure 3, CH_C 34 is verified by two different people, DVs 36 and 38, each of whom are verified by a single IV 40. CH_D 42 is also verified by 2 different people, DVs 44 and 46, but these two individuals are each verified by a different IV, 48 and 50, respectively. If we again assume that $r_{ij} = 1$ for all verifications; $\alpha = 1$ (not standardized); and $\beta = 0.5$, then the focal individuals CH_C 34 and CH_D 42 will have the same centrality scores:

$$\begin{aligned}
\text{For CH}_C: C_C &= r_{ij}[(\alpha + \beta * C_{DV36}) + (\alpha + \beta * C_{DV38})] \\
&= (1)[(1 + 0.5 * r_{ij}(\alpha + \beta * C_{IV40})) + (1 + 0.5 * r_{ij}(\alpha + \beta * C_{IV40}))] \\
&= (1)[(1 + 0.5 * 1(1 + 0.5 * 0)) + (1 + 0.5 * 1(1 + 0.5 * 0))] \\
&= 1[1.5 + 1.5] \\
&= 3
\end{aligned}$$

$$\begin{aligned}
\text{For CH}_D: C_D &= r_{ij}[(\alpha + \beta * C_{DV44}) + (\alpha + \beta * C_{DV46})] \\
&= (1)[(1 + 0.5 * r_{ij}(\alpha + \beta * C_{IV48})) + (1 + 0.5 * r_{ij}(\alpha + \beta * C_{IV50}))] \\
&= (1)[(1 + 0.5 * 1(1 + 0.5 * 0)) + (1 + 0.5 * 1(1 + 0.5 * 0))] \\
&= 1[1.5 + 1.5] \\
&= 3
\end{aligned}$$

[0081] Thus, although they are embedded in different ways in different networks, CH_C and CH_D have identical indegree Bonacich centrality scores. Nevertheless, the system is more confident that CH_D is not attempting to spoof the system. The more dispersed, less cohesive network (network 4) offers greater information non-redundancy. Information on CH_C's trustworthiness comes from three individuals (directly from 2 DVs and indirectly from a single IV), while information on CH_D's trustworthiness comes from four individuals (directly from 2 DVs and indirectly from two IVs). Everything else being equal, the system can have greater confidence in CH_D's trustworthiness.

[0082] D. Network Closure

[0083] Another phenomenon (occurring in social networks) is also relevant. Network closure measures how closely-knit a network is. That is, the degree to which its members are connected to each other. The more closed (closely-knit) a network, the more connected its members are to each other. In Figure 3, network 3 is has greater closure than network 4.

[0084] By definition, closure is closely related to information redundancy. Practically, if a network's members are highly connected with each other, their information sources are more likely to be redundant. Consequently, the greater a network's closure, the greater the information redundancy within that network.

[0085] Closure, however, also has more insidious consequences. Closure generates enforceable trust within a tightly-knit group. By definition, social groups with closure possess multiple, redundant information channels. Thus, information flows freely within the group, ensuring that everyone “in the loop” knows everything about everyone else. This spread of rumors has three converging effects. Since group members know a great deal about each other, they know what to expect from each other. Additionally, members quickly find out about people who violate social norms, and get each other to collectively punish these violators. Most importantly, members develop a collective sense of affection for the group and its members. Taken together, tightly-knit groups (with network closure) acquire substantial potential for collective action. Such enforceable trust is particularly powerful for mobilizing groups against outsiders, including authority figures. For instance, police investigators often face great difficulty investigating incidents that happen inside closed communities (e.g., cults and small ethnic groups).

[0086] A small, tightly-knit group of friends has greater capacity to spoof the system than an equal number of people who do not know each other. For instance, suppose a married man decides that he desires other women on the side. Ordinarily, on on-line dating sites, the system would mark him as a married man and hinder his efforts. But, if the man convinces four friends to vouch for the (false) fact that he is single, then he may defeat the safeguards offered by the system. In a tightly knit group it is likely that his friends would comply with this request, not only because they want to help their friend, but also because they fear social retribution from the others in the group. Here, the system is an outsider to this group and is a prime target when it gets in the group’s way.

[0087] Recognizing this potential for fraud, in embodiments of the present invention the system guards against such events by penalizing CHs who have highly-closed, egocentric networks. In other words, the greater a CH’s apparent ability to spoof the system, the less confidence the system must have in that individual’s self-assertions. While a majority of people that belong to closely-knit groups of friends may have no incentives to self-assert false attributes, the system is configured to penalize them based on their capacity (not necessarily their intention) to spoof.

[0088] But this presents a problem for systems that rely on indegree Bonacich centrality, which rewards closure instead of penalizing it. For instance, consider Figure 4: networks 5 and 6 are identical, with a CH 52 being verified by two DVs 54 and 56, each verified by a common IV 58,

except for a single DV-to-DV verification 60, present in network 6. If the two DVs 54 and 56 and the CH 52 all know each other, they are more likely to represent something like the group of friends in the above example. Thus, the system should have reduced confidence in CH 52 for the network 6 situation compared with the situation in network 5. However, indegree Bonacich centrality is higher for the network 6 case:

$$\begin{aligned}
\text{For network 5: } C_{CH} &= r_{ij}[(\alpha + \beta * C_{DV54}) + (\alpha + \beta * C_{DV56})] \\
&= (1)[(1 + 0.5 * r_{ij}(\alpha + \beta * C_{IV58})) + (1 + 0.5 * r_{ij}(\alpha + \beta * C_{IV58}))] \\
&= (1)[(1 + 0.5 * 1(1 + 0.5 * 0)) + (1 + 0.5 * 1(1 + 0.5 * 0))] \\
&= 1[1.5 + 1.5] \\
&= 3
\end{aligned}$$

$$\begin{aligned}
\text{For network 6: } C_{CH} &= r_{ij}[(\alpha + \beta * C_{DV54}) + (\alpha + \beta * C_{DV56})] \\
&= (1)[(1 + 0.5 * r_{ij}\{(\alpha + \beta * C_{IV48}) + (\alpha + \beta * C_{DV56})\}) + (1 + 0.5 * r_{ij}(\alpha + \beta * C_{IV50}))] \\
&= (1)[(1 + 0.5 * 1\{(1 + 0.5 * 0) + (1 + 0.5 * (1)(1 + 0.5 * 0))\}) + (1 + 0.5 * 1(1 + 0.5 * 0))] \\
&= 1[1 + 0.5(1 + 1.5) + 1.5] \\
&= 1[1 + 1.25 + 1.5] \\
&= 3.75
\end{aligned}$$

[0089] One solution for this dilemma is to follow the spirit of indegree Bonacich centrality by accounting for network redundancy and closure. A score is generated based on a focal individual's immediate neighbors in a social network while addressing redundancy and closure.

[0090] Various embodiments of the present invention, however, adopt a different approach. This solution disaggregates the impacts of direct (DV) and indirect (IV) verification, and, taking advantage of this disaggregation, incorporates mechanisms for rewarding CHs for greater local network non-redundancy and penalizing local network closure. This solution has two primary components: direct embeddedness and indirect embeddedness.

[0091] E. Direct Embeddedness

[0092] Direct embeddedness refers to DVs' contribution towards the system's confidence in a given CH attribute (SA). DV effects on SA have a strong resemblance to indegree Bonacich

centrality. Each DV verifying a CH attribute contributes a fraction (e.g., one-tenth) of his/her user score (SU) to the attribute's SA. This is equivalent to indegree Bonacich centrality where $\beta = 0.1$, $\alpha = 0$ and $r_{ij} = 1$.

[0093] Unlike indegree Bonacich centrality, direct embeddedness adjusts for closure. If any specified DV is verified by (or verifies) another DV, these two DVs' direct embeddedness contribution to SA is divided by 1.0. This adjustment accounts for the potential that the two DVs could collaborate with the CH to help him/her spoof the system. Consequently, the total direct embeddedness contribution to SA equals:

$$SA_i(DE) = \sum_j (\beta * SU_j / r)$$

Where $SA_i(DE)$ = the direct embeddedness contribution towards an attribute of the i^{th} CH, $\beta = 0.1$, SU_j = the SU value for the j^{th} DV verifying the relevant CH attribute, $r = 1.0$ if the j^{th} DV verifies (or is verified by) another DV, and j = total number of DVs verifying the CH.

[0094] F. Indirect Embeddedness

[0095] Indirect embeddedness refers to IVs' contribution towards the system's confidence in a given CH attribute (SA). IV effects on SA also resemble indegree Bonacich centrality, but with a crucial difference: IVs are two degrees of separation removed from the CH, not one (like indegree Bonacich centrality and direct embeddedness). Each IV verifying a DV contributes a small fraction (e.g., $1/40^{th}$) of his/her user score (SU) to a CH attribute's SA. This resembles indegree Bonacich centrality where $\beta = 0.025$, $\alpha = 0$ and $r_{ij} = 1$. However, it is important to note that j represents the set of all IVs, not DVs.

[0096] Indirect embeddedness adjusts for redundancy by limiting the total indirect embeddedness contribution per DV. Each DV (except for those that lack IVs altogether) links IVs with the CH. The IVs "belonging" to any single DV contributes a maximum number (e.g., 2) of points to SA. For instance, consider 10 IVs (each with $SU = 50$) that are connected with a CH through a single DV. Each IV contributes $1/40 \times 50 = 1.25$ points to SA, for a total of 12.5 points. However, the IVs belonging to a single DV can only contribute a number of points up to the threshold value (2 in this example), so the total contribution to the subject CH's SA is capped at that threshold (2). This reflects the intent that a single DV's local network should not have

undue influence on the CH's overall SA scores. Without this cap, a CH could elevate his/her SA scores by being verified by a single DV with a large number of IVs. This would violate a need to privilege non-redundant sources of information about CH trustworthiness.

[0097] Similarly, indirect embeddedness adjusts for redundancy by not double-counting IVs that verify two (or more) different DVs. When a single IV verifies multiple DVs, the SU score for such IVs contribute towards CH SA scores through multiple channels, one for each DV that the IV verifies. Considering that these channels are redundant and provide the system redundant information about the CH's trustworthiness, these channels should not be double-counted. To prevent double-counting, an IV's SU score is divided by the number of DVs that the IV verifies.

[0098] Indirect embeddedness, consequently, is calculated in a multistage process. For each IV, its contribution to SA is calculated by: (1) taking the IV's SU score, and dividing by a fraction (e.g., 40) and (2) dividing the result by the number of DVs the IV verifies. This creates several "score fragments" that are each (3) added to CH SA scores, (4) conditional on that particular DV's IVs contributing a total number of fragments that do not collectively exceed a threshold (e.g., the 2-point cap discussed above). For instance, an IV with $SU = 40$ that verifies four different DVs contributes $(1140 \times 40) / 4 = 0.25$ points through four different channels. Each channel is subject to the 2 point cap. If one of these channels has already exceeded that cap, only three channels (each worth 0.25 points) actually contribute to the relevant CH SA score, for a total of 0.75 points. By making sure that IVs are not double-counted in calculations, this safeguard rewards CHs whose local networks have a high degree of non-redundancy.

[0099] Overall, the total indirect embeddedness contribution to SU can be expressed as:

$$SA_i(IE) = \sum_j (\max(\sum_k \gamma * f(SU_k), 2))$$

Where $SA_i(IE)$ = the indirect embeddedness contribution towards the i^{th} CH; j = the number of DVs; k = the number of IVs associated with the j^{th} DV; $\gamma = 0.025$; $f(SU_k)$ = the SU value for the k^{th} IV associated with the j^{th} DV, divided by the number of different DVs k is associated with.

[00100] G. Embeddedness and Threats

[00101] The present system identifies threats who are trying to spoof the system (i.e., self-assert false attributes). It provides its users opportunities to report other users who are self-asserting false attributes in two different situations:

[00102] Request to Validate False Attributes: Consider a situation where user A asks user B to validate an attribute that B knows to be false. B can validate the attribute as requested, compromising the system's integrity. Conversely, B can report A for A's attempt to self-asserting false attributes. The "self-regulation through social norms model" is appropriate here.

[00103] Of course, not all users in B's situation will report A's false self-assertions. Users who are connected by a large number of redundant paths (i.e., members of a tightly-knit group with high closure) are likely to lie for each other; such users will validate (rather than report) false self-assertions.

[00104] Unlike highly closed networks, networks with low closure work to the present system's advantage. Individuals who know each other but are not connected through redundant paths have the ability to report each other. They have no friends in common. Consequently, they are not members of the same tightly-knit group, and need not worry about the consequences of violating enforceable trust. Thus, assuming that users of the present system have a desire to defend against intruders, such users have the knowledge and motivation to report false self-assertions.

[00105] H. Embeddedness in Other Social Networks

[00106] The present system is configured to award greater confidence for CHs embedded in other, on-line social networks (i.e., social networks other than the web of trust created by the present system). This is based on a recognition that a CH who is highly embedded in such other networks is more likely to be trustworthy than someone who is not. However, not all social networks are treated equally.

[00107] Relationships in some on-line social networks provide greater trustworthiness than relationships in other networks. Two mechanisms differentiate different networks. First, some networks scrutinize their users' asserted attributes more than other networks. For instance, some social networks validate their users' school and/or business affiliations by requiring e-mail addresses from the appropriate .edu and/or .com domains. Thus, within such a network a user

cannot self-assert himself/herself as a student of a particular institution without a corresponding e-mail address from that institution. Also, some social networks offer categorization of contacts within their networks and include (optional) mutual-confirmation, so that someone claiming to be colleague from a particular company must be confirmed by the user before he/she is permitted to self-assert that affiliation within his/her user profile. Networks that adopt such measures are more secure than networks lacking such mechanisms, hence data obtained from such networks is deemed to be more reliable than similar information obtained from other social networks.

[00108] It is also true that some social networks embody deeper social ties than others, based on the network's culture and purpose. For instance, some social networks are intended to provide career-related networking opportunities, while others are intended for entertainment purposes. Assuming that people are more likely to engage in frivolous activities for entertainment than career-related purposes, those networks intended for the career-related purposes are deemed to provide relationship information that is more likely to be meaningful than relationship data obtained from social networks intended primarily for entertainment purposes. This distinction can be realized through weighting factors.

[00109] Therefore, in various embodiments of the invention, credential scores receive contributions for an individual's embeddedness in social networks other than the system's web of trust, and these contributions may be based on the nature of the other social network in which the individual is involved and the number (and perhaps type) of connections the individual has within those networks. The total contribution for such embeddedness to the individuals overall SA may be capped (i.e., weighted).

[00110] I. Identity Measures

[00111] Social network analysis (SNA) measures for embeddedness (such as those discussed above) represent powerful ways to predict CH attributes' truthfulness. However, other techniques represent useful complements to SNA-based analyses. Non-SNA validation techniques (identity measures) focus on three aspects of self-assertions:

1. User profiles having a greater number of meaningfully-completed attributes (e.g., name, address, photo, multiple distinct e-mail addresses, etc.) require greater time and effort to create.

2. Users who provide difficult-to-replicate attributes or features (e.g., social network profiles with a long, consistent history of activity) cannot re-use the same attributes to create additional (fake) profiles.
3. Users who have existing profiles on certain trusted profile sites (such as the career-oriented social network sites discussed above). The principle here is that if someone has a profile on such a site and possesses contacts of a significant quantity, the present system can trust the self-assertions of this virtual person to a greater extent versus someone who does not have such an affiliation.

[00112] In other words, user profiles requiring greater effort to create, that include nonreplicable attributes and leverage other “trusted” profiles, more likely contain truthful self-assertions than profiles lacking some or all of these features. Consequently, the present system’s identity measures assign higher confidence (SA) to attributes belonging to CHs who self-assert (1) greater amounts of (2) difficult-to-generate attributes. At the same time, it is recognized that many, if not most, identity measures are easily self-asserted by strategic, determined individuals intent of spoofing; consequently, the present system weights scores obtained through such identity measures relative to scores developed through network analysis.

[00113] J. Trusted Anchors

[00114] The trusted anchor process is another useful complement to SNA-based analyses. Various entities maintain vast amounts of data concerning individuals. For instance, credit rating agencies not only possess information on peoples’ financial positions, but also their socio-demographic attributes. The present system validates trusted anchors’ self-asserted attributes against their credit reports or information obtained from similar, trusted databases (preferably on-line databases) or requires an in-person proofing of those attributes.

[00115] The trusted anchor process is less optimal than SNA processes for two reasons. First, this process involves additional “friction” for users. Document review, on-line form verification and in-person processing all create additional work for users. Second, validating users against on-line databases usually involves monetary costs. Credit agencies (and other database owners) typically will not allow access their data for free. Furthermore, many of these databases do not provide global, all-ages coverage, which makes them less than optimal sources

of information. Even if these databases are aggregated, they often contain inaccurate data which makes matching only partially automated, and often requires human-based exception handling at much higher costs. In contrast, SNA-based validation involves neither of these costs; thus, SNA may be preferable.

[00116] Yet, the trusted anchor process represents an ideal complement to SNA-based techniques. Some users may be isolates having little or no connection with the web of trust. The trusted anchor process gives these users an opportunity to validate their attributes at a much higher confidence level. Additionally, the trusted anchor process is useful for double-checking CH attributes in two situations: (1) when a CH attribute's veracity is challenged by other users; and (2) random spot-checks of members. Although the trusted anchor process is not a suitable replacement for the web of trust, it represents an excellent complement.

[00117] Trusted anchors may be granted powerful responsibilities within the present system. Through direct embeddedness, trusted anchors can influence other users' scores dramatically. Since they are given extremely high SU scores (above those which can be achieved by other users), trusted anchors contribute dramatically to SA scores for user attributes they verify. Consequently, they are implicitly made responsible for the trustworthiness of their local network as a whole. Trusted anchors also provide a powerful method to "seed" the network with highly trustworthy individuals who can then propagate their trust into the network.

[00118] K. Institution of Trust

[00119] The present system is imbued with features that create strong social norms against users self-asserting false attributes. In many respects, this principle strongly resembles the self-regulation through social norms model. However, the principle differs from its predecessor in two important ways: it is backed with (1) verification algorithms and (2) legal consequences. In other words, the system creates an enforceable version of the self-regulation through social norms model.

1. Individual vs. Group Rewards: A "conspiracy" to spoof the system may benefit a single user (e.g., a solitary sexual predator), or several different users colluding with each other (e.g., a ring of child molesters). This distinction structures potential participants'

incentives in different ways. For instance, someone who is asked to “help a friend” cheat the system is likely to respond in different ways depending on the risk he/she will incur.

2. Punishment: A related question is the need for secrecy. On one hand, potential threats require secrecy because they aim at deceiving other users of the system. On the other hand, potential threats maintain secrecy because they fear punishment for their misdeeds.

Together, these two dimensions constitute a 2x2 typology of potential threats, as shown in Table 1:

		Benefits Accrue To:	
		Individual	Group
Punishment if Caught	Severe	(1) Solitary: benefiting individual acts alone, as incentive structure prevents him/her from enlisting compatriots.	(2) Conspiracy: potential beneficiaries use “honor among thieves” (mutual trust) to achieve shared malfeasance.
	Negligible	(3) Help-a-Friend: benefiting individual enlists non-benefiting compatriots (who have little to lose).	(4) Just-for-Fun: potential beneficiaries enlist each other (and non-participating friends) to achieve shared malfeasance.

[00120] Case I (Solitary Threats): Where (1) potential punishments are severe, and (2) benefits accrue to single individuals, the threat is likely to consist of a single individual unable to enlist compatriots. The benefiting individual has the incentive to incur substantial risks. However, his friends (or other accomplices) have no reason to help him in the face of harsh potential punishments. Consequently, such threats are less dangerous than other types of threats (see below). For instance, a highly-motivated child molester might self-assert that he is an 11-year old. However, this assertion cannot obtain a high confidence score (SA) because the associated user cannot attempt to obtain verification of this (false) attribute by other users for fear of being reported by these other, who have no incentive to help him.

[00121] Case 2 (Conspiracy): Where (1) potential punishments are severe, and (2) benefits accrue to multiple individuals, the threat is likely to consist of a group of closely-knit conspirators bound together by enforceable trust. Having pre-existing, redundant social relationships, these conspirators have “honor among thieves”, i.e., the mutual trust required to cooperatively pursue illegal activities. Such threats are likely to resemble a child molester ring,

where several molesters band together to represent one of their members as a minor.

Conspiracies are likely to come in two varieties: unintelligent conspirators, who attempt to perpetrate frauds and are caught (e.g., on the basis of records maintained by the system), and intelligent conspirators, who recognize the risks and abandon attempts to spoof the system.

[00122] Case 3 (Help-A-Friend): Where (1) potential punishments are negligible, and (2) benefits accrue to a single individual, the threat is likely to consist of the benefiting individual and a group of his/her friends possessing high network closure. Without facing potential punishments, the threat's friends have an incentive to help their friend or face the collective wrath of the group (through enforceable trust). Although such threats are difficult to defend against, the stakes are considerably lower (assuming that punishments are correlated with the severity of a "crime").

[00123] Case 4 (Just for Fun): Where (1) potential punishments are negligible, and (2) benefits accrue to a group, the threat is likely to consist of that group. Without facing potential punishments, this group has an incentive to collectively spoof the system that is not countered by fear of punishment. Like case 3, such threats are low-risk but difficult to defend against. For example, consider a group of 13-year old children self-asserting that they are 18, perhaps to get around something like an age-restriction at a certain web site. These individuals do not harm anyone (except perhaps themselves) by their fraud. Such threats are extremely likely to avoid spoofing behaviors, however, if they face consequential legal sanctions.

[00124] These above case scenarios illustrate the need to back up the on-line system with physical-world punishments, including but not limited to strict penalties for violations of terms of service. Abusers (including those who would falsely verify assertions of a CH) may also be deterred by conducting credit checks on all users, and performing random verifications of user information against credit reports. Through such a strategy, the system establishes and maintains a reputation for being intolerant of users who self-assert false attributes. Consequently, the system obtains the benefits of the "self-regulation through social norms model" and backs it with enforcement mechanisms. Through these measures, the system establishes itself as an institution of trust and at the same time reduces the number of false positive verifications occasioned by people verifying attributes without actual knowledge of the CH.

[00125] In addition, the present system may incorporate “user feedback” in the sense that users can report falsehoods which they uncover about others (e.g., invalid self-asserted ages, marital status, etc.). Following appropriate investigations and verifications of these inaccuracies, individuals responsible for the inaccurate assertions, including perhaps verifiers responsible for collusion or negligence, can be punished. As these investigations identify threat vectors, the system can be modified to eliminate same.

[00126] L. Computing a Credential Score

[00127] The present methods and systems thus involve a number of techniques for increasing trust between users as indicated in Table 2:

Table 2

Mechanism	CH attribute validation through:
Direct Embeddedness*	Embeddedness in the system’s web of trust (direct)
Indirect Embeddedness*	Embeddedness in the system’s web of trust (indirect)
Embeddedness and Threats	Reporting threats embedded in the system’s web of trust
Embeddedness in Other Social Networks*	Embeddedness in other social networks
Identity Measures*	Verification using non-network measures
Trusted Anchors*	Verification using existing (on-line) databases
Institution of Trust	Cultural/institutional construction and enforcement

In various embodiments of the invention, some of these measures (marked with * in Table 2) are synthesized into a single SA score for a CH.

[00128] In some embodiments, the system’s response to threats are not so synthesized into the SA score. Consider for example, a situation where one user reports another user’s self-asserted attributes as false, but no definitive resolution of the assertion either way can be made using objectively verifiable data (e.g., from publicly available database sources). Under these circumstances, no objectively quantifiable demerits can be incorporated in the subject SA. Hence, the system reports demerits separately from the SA score, possibly with explanations of the dispute, allowing an RP to make an independent judgment of the situation. Over time, some of these situations may be verified through the trusted anchor process, allowing the demerits to be incorporated in the SA score (or eliminating them as false challenges).

[00129] Finally, the system exists as an institution of trust. Such an institution does not validate individual users’ scores; rather, it enhances trustworthiness. Thus, it is not appropriate to incorporate this mechanism into SA score calculations.

[00130] In one embodiment of the invention, the single SA score is synthesized as follows: (1) each contributing mechanism from Table 2 is assigned a certain number of total points which it can contribute to an overall score (e.g., this amounts to a weighting factor); and (2) the actual points attributable to the individual mechanisms (up to their respective maximum point values) for a given CH's SA are added together. Thus, SA scores are calculated through a scoresheet approach, where each mechanism is allocated a specific number of scoresheet points and the SA scores are simply the summed total of these scoresheet points. An example of such a scoresheet is shown below in Table 3.

Table 3

Mechanism	Calculation	Maximum Points
Direct Embeddedness*	$\sum_j (\beta * SU_j / r)$	10
Indirect Embeddedness*	$\sum_j (\max(\sum_k \gamma * f(SU_k), 2))$	30
Embeddedness in Other Social Networks*	Threshold: If (# of contacts in other network > 20, 5, 0), etc.	5
Identity Measures*	Baseline score (e.g., 10 points)	5*
Trusted Anchors*	Baseline score (e.g., 50 points)	50
Total		100

* identity measures substitute for embeddedness in other social networks, thus, this mechanism's points are not cumulative.

Any points generated by a mechanism in excess of the maximum number of its assigned scoresheet points are truncated (ignored).

[00131] This scoresheet has several noteworthy characteristics:

1. A user can reach 100 points maximum. However, to exceed 50 points, the user must become a trusted anchor.
2. Indirect embeddedness accounts for the majority (60%) of the remaining 50 points. To generate a large number of indirect embeddedness points, CHs must be connected with a large number of IVs. Assuming that potential spoofers will have difficulty creating a large number of fake profiles, the indirect embeddedness measure is exceedingly difficult to spoof. Consequently, it is given the greatest weight in the scoresheet.

3. Direct embeddedness accounts for a substantial proportion (30%) of these points. On one hand, CHs require several different DVs to generate many direct embeddedness points. On the other hand, the number of DVs required is low enough to be spoofed by an extremely dedicated spoofer. For instance, a spoofer might create 10 fake accounts, each with maximum identity measures ($SU = 10$), that each validate an 11th account that already has 10 identity points. Thus, the spoofer is able to create an account with 20 points. The reason that direct embeddedness points are capped at 15 is to prevent spoofers from reaching higher point values through this mechanism.
4. Embeddedness replaces identity measures whenever possible. In some networks, embeddedness is much more difficult to replicate than identity measures, which are strictly self-asserted.

[00132] According to another embodiment of the invention, SA scores are replaced with percentage likelihoods that a self-asserted attribute is actually true. In either instance, the SA score (or the likelihood determination) may be reported to an RP upon request. For example, the RP may be a web site intended for adults. When a user attempts to access the web site and reports his/her age and another identifier (e.g., an e-mail address), the web site may send a request to the system to report the SA for the subject individual's (identified by the e-mail address) age. Here, age would be the attribute under test and the SA for the age would be computed as the sum of the contributing mechanism scores. It would then be up to the subject web site to admit the user or deny entry (e.g., on the basis of whether or not the reported SA for the user's age met or exceeded a required threshold).

[00133] Thus, methods and systems for verifying on-line identities and, more particularly, attributes of such identities, using social network analysis and other means have been described. The examples presented in connection with this description were intended merely to illustrate aspects of the present invention, and should not be read as limiting the invention. For example, embodiments of the present invention find application in connection with micro-credit lending programs. It is known that many people in the Third World do not have established credit histories, at least not with well-known credit rating agencies which lenders look to for reports on credit worthiness. Thus, many micro-credit lending agencies, which have become popular among

Internet users, are having a hard time identifying creditworthy versus non-creditworthy individuals. The present invention can be used to alleviate this situation.

[00134] By replacing “confidence in identity attributes” by “confidence that someone will repay a loan,” the present invention provides a means for an individual to evaluate whether or not to extend credit (e.g., in the form of a loan) to another. Individuals without established credit histories can now be vouched for by other individuals who have established credit histories. The pattern of these verifications can be analyzed in the same manner as identity verifications discussed above. In such a scenario, the CH is the individual seeking credit (or a loan), DVs and IVs are individuals with established credit histories, and the RP is the putative lender. In some instances, the micro-lending instantiation may require some modifications to the above-described processes; for example, examining how a default would affect both the borrower and the individuals vouching for the borrower, and modifying the non-network analyses accordingly (e.g., by ascribing different weightings to same).

[00135] Further, from the above description, it should be apparent that various embodiments of the present invention may be implemented with the aid of computer-implemented processes or methods (a.k.a. programs or routines) that may be rendered in any computer language, stored on any tangible computer-readable medium, and executed by a computer processor in order to perform the intended functions described above. Where reference was made to algorithms and symbolic representations of operations on data, such operations may be made on data stored within a computer memory or other tangible computer-readable medium. These algorithmic descriptions and representations are the means used by those skilled in the computer science arts to most effectively convey the substance of their work to others skilled in the art. Thus, throughout the description of the present invention, use of terms such as “processing”, “computing”, “calculating”, “determining”, “displaying” or the like, were intended to refer to the action and processes of a computer system, or similar electronic computing device, suitably programmed to manipulate and transform data represented as physical (electronic) quantities within the computer system's registers and memories into other data similarly represented as physical quantities within the computer system memories or registers or other such information storage devices in order to implement the above described processes. Thus, such a computer system under these programming conditions is best viewed as an apparatus specially configured to implement the present methods.

[00136] An advantage of the computations of direct and indirect embeddedness discussed above, when instantiated as computer-implemented processes, is that they can be run in linear-time (i.e., n -time in Big-O notation) for most on-line social networks. In contrast, most social network-based algorithms do not run in linear time. Because the present computations run more quickly than $n \log n$ time, it is scalable to large-scale applications. To better appreciate this point, consider that an algorithm that runs in n^2 time may be run for 100 users without much difficulty. To run the same algorithm for 1000 users, however, 100 times the computing power is required because the computational needs increase exponentially. The same increase, from 100 to 1000 users, would only require a 10 time increase in computing power for a linear algorithm such as that provided by the present invention.

CLAIMS

What is claimed is:

1. A computer-implemented method, comprising reporting, in response to receiving a request therefor, a credential that represents an estimate as to how likely a self-asserted attribute of an individual representing said attribute as true is in fact true, wherein the estimate is computed through a plurality of mechanisms, including an examination of a web of trust within which the individual is embedded and non-network analysis based measures of a veracity of the attribute's asserted value.
2. The method of claim 1, wherein the examination of the web of trust includes computing a contribution for embeddedness of the individual in the web of trust.
3. The method of claim 2, wherein the examination of the web of trust includes computing contributions for direct embeddedness of the individual in the web of trust and indirect embeddedness of the individual in the web of trust.
4. The method of claim 1, wherein the non-network analysis based measures include identity measures which reward the individual for association with user profiles including difficult to replicable elements.
5. The method of claim 1, wherein the non-network analysis based measures include verification of the attribute with information obtained from trusted sources outside of the web of trust.
6. The method of claim 1 wherein the estimate is computed using weighted contributions for direct embeddedness of the individual in the web of trust, indirect embeddedness of the individual in the web of trust, embeddedness of the individual social in networks other than the web of trust, identity measures which reward the individual for association with user profiles including difficult to replicable elements, and verification of the attribute with information obtained from trusted sources outside of the web of trust.
7. The method of claim 2, wherein contributions for direct embeddedness of the individual in the web of trust are determined according to a computation of the individual's modified indegree Bonacich centrality within the web of trust.

8. The method of claim 2, wherein contributions for indirect embeddedness of the individual in the web of trust are determined according to a computation of the individual's modified indegree Bonacich centrality within the web of trust modified so as to limit a total indirect embeddedness contribution per verifying member of the web of trust for the individual.
9. The method of claim 8, wherein contributions for indirect embeddedness are capped at a threshold.
10. The method of claim 1, wherein the estimate is computed through a scoresheet approach in which the individual mechanisms by which trustworthiness of the self-asserted attribute is measured are each allocated a specific number of scoresheet points and a credential score is a summed total of the scoresheet points.
11. The method of claim 10, in which contributions to the credential score for indirect embeddedness of the individual in the web of trust comprise a majority of the scoresheet points for the examination of a web of trust within which the individual is embedded.
12. The method of claim 10, wherein contributions to the credential score attributable to verification of the attribute with information obtained from trusted sources outside of the web of trust comprise a single largest component of the scoresheet points.
13. A computer-implemented method, comprising quantitatively measuring an individual's embeddedness within a social network and assigning a score thereto, combining said score with a quantitative measure of a veracity of the attribute's asserted value as determined through non-network based analysis to produce a combined score, and reporting said combined score as a measure of trustworthiness of a self-asserted attribute of the individual.
14. The method of claim 13, wherein measuring the individual's embeddedness within the social network includes determining contributions for the individual's direct embeddedness and indirect embeddedness in the social network.
15. The method of claim 14, wherein a contribution for the individual's direct embeddedness in the social network is determined by computing the individual's modified indegree Bonacich centrality within the social network.

16. The method of claim 13, wherein a contribution for the individual's indirect embeddedness in the social network is determined by computing the individual's modified indegree Bonacich centrality within the social network, wherein modification limits a total indirect embeddedness contribution per verifying member of the social network for the individual.

17. The method of claim 13, wherein the non-network analysis includes a quantitative contribution for identity measures indicative of the individual's association with user profiles including difficult to replicable elements.

18. The method of claim 13, wherein the non-network analysis includes verification of the attribute with information obtained from trusted sources outside of the social network.

19. The method of claim 13, wherein the combined score is computed through a scoresheet approach in which each quantitative measure is allocated a contribution to the combined score up to a threshold.

20. A computer-based method, comprising determining a quantitative measure of a trustworthiness of a self-asserted attribute of an individual through a combination of analysis of a social network of which the individual is a member and non-network based analyses, and reporting said measure.

21. A computer-based method, comprising determining a quantitative measure of a likelihood that an individual will repay a loan through a combination of analysis of a social network of which the individual is a member and non-network based analyses, and reporting said measure.

ABSTRACT

Self-asserted socio-demographic attributes of individuals' identities are verified using social network analysis and other means. Through these processes, parties to a transaction or interaction are provided a measure of confidence about another party's self-asserted socio-demographic attributes, such as age, gender, marital status, etc., in order to assist in determining whether or not to pursue the transaction or interaction. The measure of confidence may be provided as a quantitative "score" indicative of the likelihood the user's self-asserted attribute is actually true. The quantitative score is derived by analyzing a web of trust in which the user is embedded.

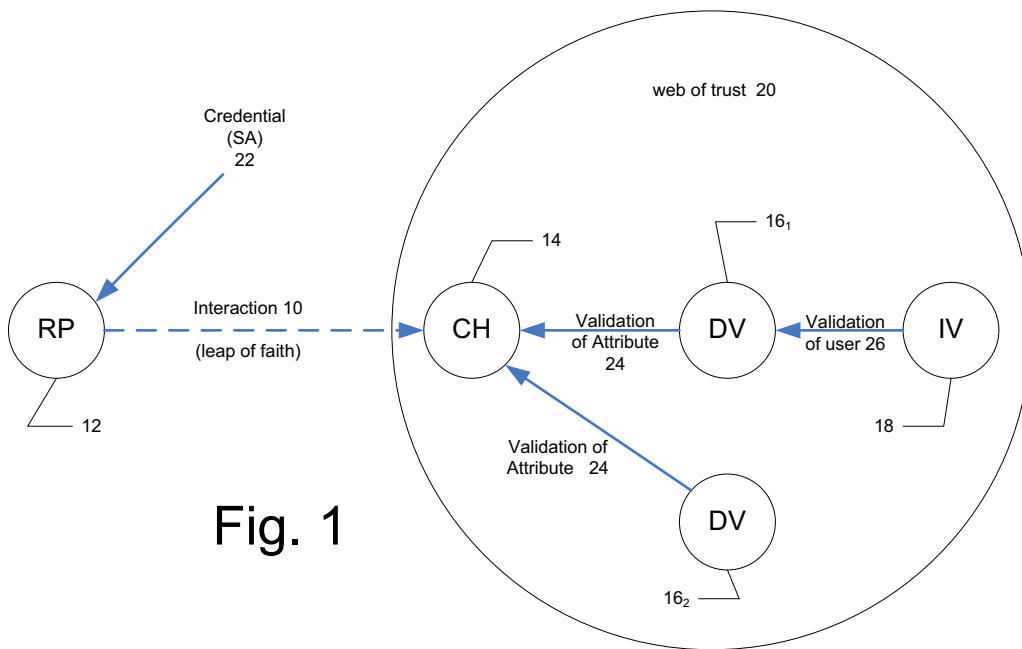


Fig. 1

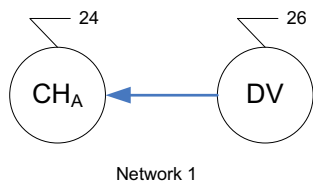


Fig. 2

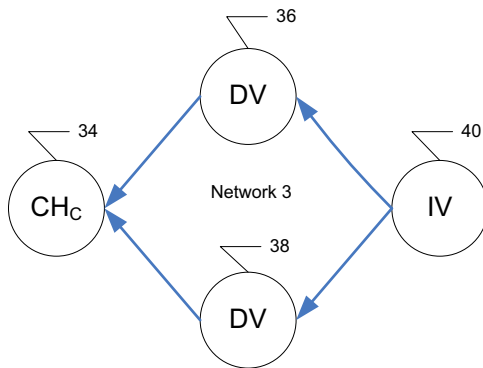
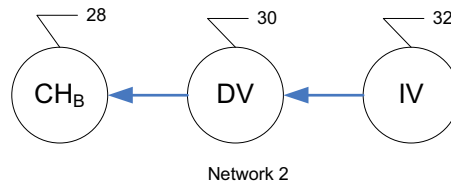


Fig. 3

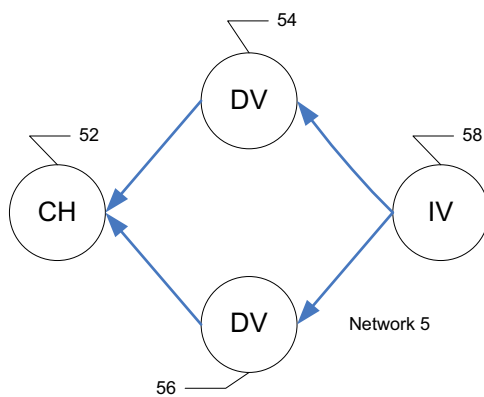
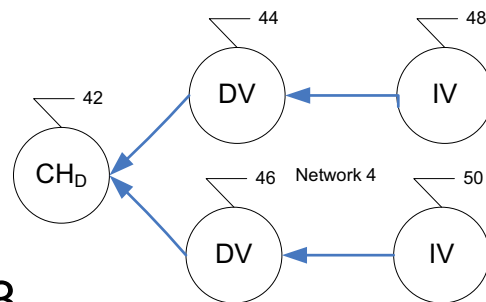


Fig. 4

