Out of the Wild: On Generating Default Policies in Social Ecosystems

Imrul Kayes
Computer Science and Engineering
University of South Florida
Tampa, Florida USA 33620
Email: imrul@mail.usf.edu

Adriana Iamnitchi
Computer Science and Engineering
University of South Florida
Tampa, Florida USA 33620
Email: anda@cse.usf.edu

***Abstract*—Combining and incorporating rich semantics of user social data, which is currently fragmented and managed by proprietary applications, has the potential to more accurately represent a user's social ecosystems. However, social ecosystems raise even more serious privacy concerns than today's social networks. This paper proposes to model privacy as contextual integrity by using semantic web tools and focuses on defining default privacy policies, as they have the highest impact.**

***Index Terms*—Privacy as contextual integrity; social ecosystems; online privacy**

I. Introduction

Social ecosystems [1] refer to the aggregation of rich datasets of user-to-user interactions in support of social applications. This data is collected from Internet-mediated social interactions (such as declared relationships in online social networks or tagging/contributing content in user-generated content platforms), from public profiles (to infer homophily relationships), and from phone-recorded real-life interactions (such as co-location sensing and activity identification). Social ecosystems have enabled a large set of social applications in various domains such as recommender systems [2], email filtering [3], and detecting conflicts of interest [4].

User privacy in online activities is already a hot issue due to lack of formal framing [5]. The primary aspect of social ecosystems, that of aggregating data from various sources to provide it (possibly processed) to a diversity of applications, significantly amplify the privacy concern. First, aggregated data from different contexts of activity presents a more complete and possibly uncomfortable picture of a person's life. Second, data is to be exposed to a variety of applications, themselves from different contexts of activity, from personal to professional.

Numerous solutions addressed privacy in social ecosystems, typically in the context of a particular system [6], [7] or for particular application scenarios [8]. Little addressed, however, is the setting of a *default* privacy policy. While users are invited to change the default privacy settings, in reality very few do it. For example, more than 99% Twitter users retained the default privacy setting with their name, list of followers, location, website, and biographical information are visible [9]. Another study [10] on a college network shows that a majority (87% on average) of students have default or permissive privacy settings in Facebook.

The privacy challenge is fundamentally due to the lack of a universal framework that establishes what is right and wrong [5]. Nissembaum proposed such a framework in her view of privacy as contextual integrity [11]. Existing solutions to formalize this framework have adopted logic reasoning techniques [12]. The solution in [12] is a generic privacy expectation of personal information expressed in a formal language, and it is not focused to any system such as social ecosystems.

In this work we employ semantic web techniques to adopt Nissembaum's framework for defining application and platform-independent default privacy settings. To this end, we propose an extensible, fine-grained privacy model for social ecosystems based on semantic web technologies. The model implements the basic concepts of Nissembaum's privacy framework: social contexts, norms of appropriateness, and norms of information flow. It builds on ontologies used to encode social data and implicitly represent social contexts, and on RDF statements/SPARQL queries to define and verify access to data.

The contributions of this work can be summarized as follows.

- We propose an ontology-based social ecosystem data model to capture user social data aggregated from an unrestricted set of sources (e.g. Facebook, LinkedIn, etc.).
- We employ semantic web technologies to generate default privacy polices based on Nissembaum's contextual integrity theory. These polices are extensible, fine-grained and expressive enough to be changed by the user. Furthermore, the policy model is generic enough to be used in a proprietary system.
- We provide an architecture and the prototype implementation of our privacy model that automatically enforces access control policies on queries submitted to a social ecosystem knowledge base.

The rest of the paper is organized as follows. Section II introduces the contextual integrity theory, and discusses its relevance to social ecosystems. Section III describes the privacy model, system assumptions, and an architecture. Section IV presents our ontology-based data model. We present our policy specification and prototype implementation in Section V. Section VI reviews related literature and Section VII concludes the paper.

II. Privacy as Contextual Integrity in Social Ecosystems

While notoriously difficult to define [11], privacy is understood as an individual's right to determine to what extent her data shall be communicated to others. Instead of defining the term, Nissembaum proposes a reasoning framework for privacy as contextual integrity [11], where privacy is seen as neither a right to secrecy nor a right to control, but a right to appropriate flow of personal information. Nissenbaum's account of privacy as contextual integrity is based on two non-controversial facts. First, every transfer of personal information happens in a certain social context and all areas of life (and online activity makes no exception [5]) are governed by context-specific norms of information flow. For example, in a medical context patients share their physical condition with the physician but not vice versa. Second, people move among a plurality of distinct contexts, thus altering their behavior to correspond with the norms of those contexts, aware to the fact that information appropriately shared in one context becomes inappropriately shared into a context with different norms.

Two types of norms maintain contextual integrity: norms of appropriateness and norms of distribution. *Norms of appropriateness* circumscribe the type of information about persons that is appropriate to reveal in a particular context. Implemented in social ecosystems, this type of norm specifies *where* context-specific data can be communicated. For example, if Alice is a colleague of Bob in the professional context, then requests from Alice regarding Bob's gaming context such as the games owned by Bob should be denied, as the requests do not comply with the norms of appropriateness. *Norms of distribution* cover the transfer of information from one user to another. In a social ecosystem, the norm of distribution suggests a default policy that restricts the distribution of information that are shared. For example, if Alice and Bob have a shared content, then a request from Charlie to Alice regarding the content will not succeed without Bob's consent.

III. System Model and Architecture

Our general architecture fits the Social Hourglass infrastructure [13], where social sensors (Figure 1) initiate the process of collecting and transforming social signals into domain-specific social knowledge. A social sensor is an application running on behalf of a user and observing one particular social signal (for example, Facebook interactions of the user with other users) that reports processed social data to the user's aggregator. Sensor operations are context specific; a sensor is responsible for extracting data from a domain based on an ontology. For example, a LinkedIn sensor observes its user's professional data and a Facebook sensor observes the user's friendship data based on the ontology shown in Section IV.

The aggregator acts as the user's personal assistant and is responsible for another level of social data processing and sensor management (installation, configuration, etc.). It sends processed social data to Social Data Management Layer in the form of labeled, weighted social edges. User social data, extracted and aggregated from various sources, is stored in the *Social Ecosystems Knowledge Base (SEKB)*, managed by the Social Data Management Layer. (The exact architecture and design requirements for sensors and aggregators, presented in [13], are not necessary for explaining the present work.)

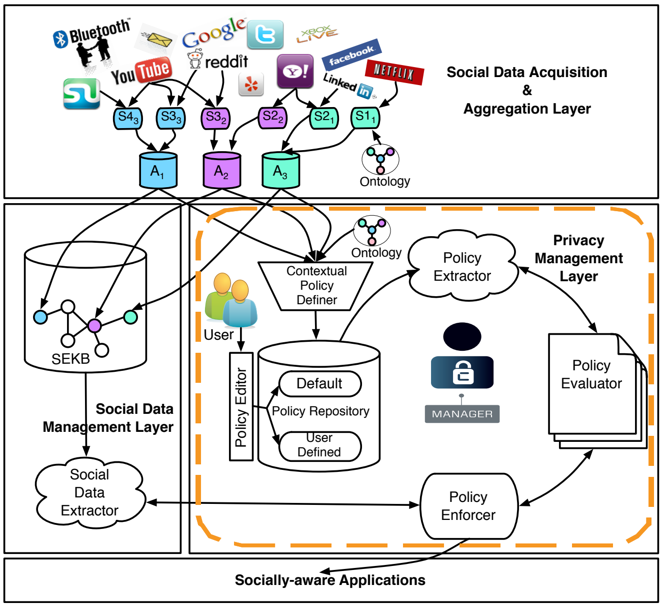

Fig. 1: A layered architecture of social data collection, personalization and management for socially aware applications along with privacy management layer.

The Privacy Management Layer in Figure 1 is responsible for managing and enforcing privacy policies, and thus for extracting and applying the default policies as well. This component communicates with the Social Data Management Layer which implements social contexts and roles.

Similar to the Dey et al.'s definition of a context [14], we define the social context of a user as the collection of social information that describes the user in a domain. For example, data about Bob's education, skills, and LinkedIn connections describe Bob's professional context. The social ecosystem of a user is the aggregation of the user's online social contexts.

Our system model is defined by the following:

1) there is an unrestricted set of disjoint social contexts in the system;
2) at any time, a user belongs to only one social context;
3) a user can have one or more roles in every social context he is part of;
4) each piece of data (resource) is assigned to only one context; however, users can share a resource with other users, case in which the resource is replicated in each of the other users' current contexts.
5) a request for a resource is made on behalf of the requester's role in the particular context in which the

requester is when the request is made;

6) a request specifies an action, which could be *read*, *write*, *delete* or *replicate* to another user's ownership.

Implementing contextual integrity in the default privacy policies requires implementing the norms of appropriateness and distribution in this system model.

The *Contextual Policy Definer* generates default access control policies based on the ontology and contextual integrity and stores them in the *Policy Repository*. Specifically, the Contextual Policy Definer generates default policies based on the following rule: only roles in a user's social context are allowed access to the user's data associated with that particular context. An example of a default policy extracted with this rule is the following: all users with a *Colleague* role in Bob's *Professional* context can access (all) his data associated to the *Professional* context. Our policy model is granular; it defines a policy for every resource covering all the contexts a user could possibly reside.

Policy Repository is a storehouse of policies and contains automated default polices reported by contextual policy definer or user defined policies formulated by user. A graphical user interface termed as *Policy Editor* provides visualization of the policies defined either by the system or user. The policy editor hides technical details of policy enforcement representation and provides a convenient and user-centric view of policies.

The system depends on a *Policy Manager*, which consists of extractor, evaluator and enforcer for handling access requests. In particular, any tentative social data request is intercepted by the *Policy Enforcer*, which in conjunction with the *Policy Extractor* and the *Policy Evaluator* decides whether the access is permitted. Permitted access requests are finally fulfilled by returning triples from social knowledge base through the social data extractor. The policies are stored in the policy repository and policy extractor extracts policies from policy repository for evaluation. The *Policy Evaluator* adds temporary information regarding the access request to it's local knowledge base, combines it with extracted policy reported from policy extractor and evaluates the policy upon request from policy enforcer.

IV. Social Ecosystems Data Model

Our social ecosystems data model, implemented as the SEKB block in Figure 1, is based on an ontology. An ontology is a set of entities, instances, functions, relations and axioms, and is used as a vocabulary for representing the knowledge of a domain.

The advantages of using ontologies in defining social ecosystems data model are multifold. First, an ontology provides a common vocabulary for social ecosystems which ensures formal and structured representation of user's contextual data. Social sensors could be employed to collect domain-specific data using the vocabulary and a richer knowledge aggregation is possible from users digital world.

Second, an ontology-based social ecosystems data model gives semantic interoperability, thus aggregated data could be exported and used in other systems. Report shows that solving semantic issues take between 40% and 80% of application integration effort, and it requires significant human intervention [15]. So, domain as well as systems independent socially aware applications will get an edge from a common vocabulary as semantics of the data is also available.

Third, high-level logic inference is possible as the data model should have semantics associated with it. For example, if Bob has contents in professional contexts (contents subClassOf ProfessionalContext) and recommendations are contents, then the inference is possible that recommendations belong to Bob's professional contexts.

Finally, a large scale social ecosystems ontology could be built by incrementally adding different context ontologies. We can also reuse existing web ontologies from different domains to meet the demand of an exhaustive scale social ecosystems ontology.

We use OWL - Web Ontology Language in modeling social contexts. OWL is more expressive than other ontology languages such as RDFS. Moreover, DAML+OIL, a richer language has been taken as the starting point for the W3C Web Ontology Working Group in defining OWL.

Figure 2 shows a sample data model of social ecosystems considering three contexts of user's digital world: professional, friendship and gaming. The representation of the ontology is *person* centric which gives a user oriented viewpoint of the data model. The three large circles are the contexts; each circle encodes context-specific knowledge and the first level contexts are subclass of the context. The set of contexts included in our model is non-exhaustive. However, they could be extended and a large scale ontology generation is possible simply by adding more contexts of a person.

Roles are modeled as relationships: for example, *isColleagueOf* in Alice's ecosystem specifies that Bob has the role of a colleague in her professional context. Roles, as relationships, are thus asymmetrical: Charlie might be a follower in Alice's followers ecosystem but Alice might not be Charlie's follower.

V. Policy Specification and Prototype Implementation

A policy is defined as a set of RDF statements. As shown in the architecture, the contextual policy definer generates policies that obey the two information norms of contextual Integrity: norms of appropriateness and distribution. Let us explain a policy generated by the policy definer for the resource *groups* in the *Professional* context : Bob's colleagues can read his professional group involvement in the *Professional* context. The policy can be formalized as the following SPARQL query(<Policy>, the prefixes p: and se: represent the namespace of policy model and social ecosystems model respectively):

```
<Policy>
ASK
where {
 ?req rdf:type p:requestor.
```

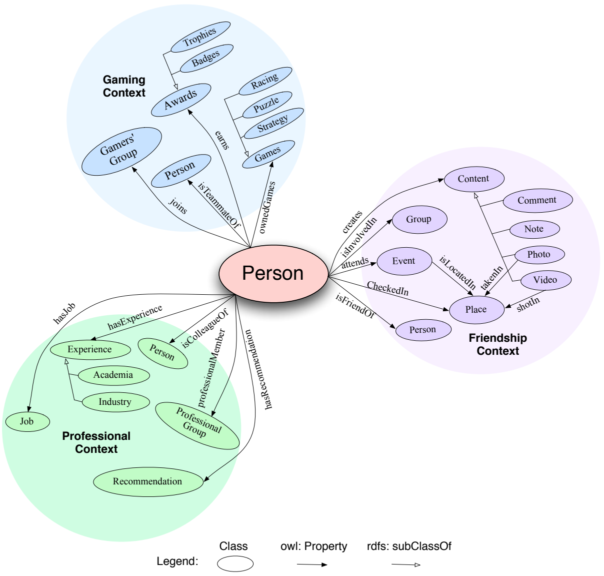

Fig. 2: A partial definition of social ecosystems ontology considering professional, friendship and gaming context.

```
 ?req p:allowed p:read.
 p:read p:performedOn Bob.
 ?req se:isColleagueOf Bob.
 Bob se:professionalMember ?group.}

<Augmented Policy>
ASK
where {
  Alice rdf:type p:requestor.
  Alice p:allowed p:read.
  p:read p:performedOn Bob.
  Alice se:isColleagueOf Bob.
  Bob se:professionalMember ?group.}
```

When a request such as "Alice wants to see Bob's professional group involvement" comes to the policy manager, the predefined policy variable `?req` will be replaced by Alice as shown by the augmented policy. The policy evaluator will temporarily insert policy-related auxiliary RDF statements such as the first three statements of the augmented SPARQL query to the knowledge base and executes the query over the modified knowledge base. The above policy representation states that the access request will be granted if Alice and Bob are colleagues. The same access request from Bob's teammate in *Gaming* context will be denied because of lack of appropriate triples, thus implementing the norm of appropriateness. Similarly, the system will disallow access to a resource that is shared or co-owned with someone, upholding the norms of distribution. For example, the following policy restricts the access to Bob's photos if they are shared.

```
<Policy>
ASK
where {
 ?req rdf:type p:requestor.
 ?req p:allowed p:read.
 p:read p:performedOn Bob.
 ?req se:isFriendOf Bob.
 Bob se:hasPhoto ?photo.
 ?photo se:status se:notShared}
```

We have implemented a prototype of Aegis in Java Platform Standard Edition 6 (Java SE 6). We use capabilities offered by Jena to implement both knowledge base and policy manager. Jena is a framework for building semantic web applications, which provides a collection of tools and Java libraries to develop semantic web and linked-data apps, tools and servers. At present Jena is the most comprehensive framework to manage RDF and Web Ontology Language (OWL) data in Java applications as it provides APIs for RDF data management, an ontology API for handling OWL and RDFS ontologies and a query engine compliant with the SPARQL specification. In our current implementation, Jena uses the file system as backing store.

VI. Related Work

As social applications becoming more popular, in recent years we have seen different solutions have been proposed to control access to users data on social networking applications.

A first category of solutions extends trust-based access control policies, inspired by research and developments in trust and reputation computation in social networks. Kruk [16] proposes Friend of a friend (FOAF)-Realm, an ontology based access control mechanism. More nuanced and complete trust related access control models are [17], [18]. While these approaches are focused on subjective (trust is always difficult to define) realization of systems, our approach is more factual; based on capturing the information semantics using an ontology-based access control policy.

Semantic rule based policies [19], [20], [21] have also emerged as a promising choice to control access to users social data. Rule based policies represents the social knowledge base in an ontology (e.g., OWL) and defines policies as Semantic Web Rule Language (SWRL) rules[1]. Access request related authorization is provided by reasoning on the social knowledge base. The drawbacks of rule based privacy models are multifold. First, authorization is provided by reasoning the whole knowledge base, thus the system is inherently centralized. Second, all the authorizations must be recomputed if a change is occurred in social knowledge base. And finally, study [22] shows that in rule based systems, rule management is a complex task and requires a team of expert administrators. In our approach the social knowledge base could be distributed, where a user's trusted peer could handle his social data request. Our social knowledge base need not to be necessarily centralized as query processing could be done using a portion of the data store (where a user has his data). Furthermore, re-computation of all policies are not required, if knowledge base changes.

[1] http://www.w3.org/Submission/SWRL/

Role and Relationship-Based Access Control (ReBAC) [23], [24] are another types of privacy models that employ roles and relationships in defining privacy policies. The conceptually closest work to this paper is probably PriMa [25]. PriMa also auto generates access control policies for users acknowledging the fact that it is perhaps not wise to rely on regular users to manually set up their access control policies because of the growing complexity of the social network and diversity of user contents. The policies in PriMa are generated based on intuitive factors such as average privacy preference of similar and related users, accessibility of similar items in similar and related users, closeness of owner and accessor (measured by the number of common friends), popularity of the owner (i.e., popular user has sensitive profile items) etc. Access control policies for profile items are finally generated aggregating these factors. The problem of this approach is that the policies are highly volatile, they change frequently based on the factors. Moreover, involvement of a lot of factors and their parametrized tuning should contribute to higher policy generation and enforcement time. As the scheme lacks implementation and evaluation, these problems might make the solution infeasible.

Our privacy model differs from the all represented above in that we are focused on generating default policies for a social ecosystem that deals with users aggregated social data from different domains, while existing solutions work only for single application scenarios. Moreover, most of those solutions did not consider default policy generation as a primary goal. Although privacy on aggregated social data was our primary focus, our policy framework is generic and expressive enough to be used in a single proprietary system also.

VII. Summary

In this paper, we proposed a semantic web standard enabled privacy model for users' social ecosystems that empowers the individual users with default fine-grained access control policies on their related information. In future, we plan to evaluate performance of privacy engine in executing access control policies while receiving socially aware requests over a large dataset.

Acknowledgements The US National Science Foundation supported this research under grants CNS 0952420 and CNS 0831785.

References

[1] A. Toninelli, A. Pathak, A. Seyedi, R. S. Cardoso, and V. Issarny, "Middleware support for mobile social ecosystems," in *COMPSAC Workshops*, 2010, pp. 293–298.

[2] P. Avesani, P. Massa, and R. Tiella, "Moleskiing.it: a trust-aware recommender system for ski mountaineering," *International Journal for Infonomics*, 2005.

[3] J. Kong, B. Rezaei, N. Sarshar, V. Roychowdhury, and P. Boykin, "Collaborative spam filtering using e-mail networks," *Computer*, vol. 39, no. 8, pp. 67 –73, aug. 2006.

[4] B. Aleman-Meza, M. Nagarajan, C. Ramakrishnan, L. Ding, P. Kolari, A. P. Sheth, I. B. Arpinar, A. Joshi, and T. Finin, "Semantic analytics on social networks: experiences in addressing the problem of conflict of interest detection," in *Proceedings of the 15th international conference on World Wide Web*, ser. WWW '06. New York, NY, USA: ACM, 2006, pp. 407–416. [Online]. Available: http://doi.acm.org/10.1145/1135777.1135838

[5] H. Nissenbaum, "A contextual approach to privacy online," *Daedalus*, vol. 140, no. 4, pp. 32–48, 2011.

[6] L. Banks and S. Wu, "All friends are not created equal: An interaction intensity based approach to privacy in online social networks," in *Computational Science and Engineering, 2009. CSE '09. International Conference on*, vol. 4, aug. 2009, pp. 970 –974.

[7] R. Baden, A. Bender, N. Spring, B. Bhattacharjee, and D. Starin, "Persona: an online social network with user-defined privacy," in *Proceedings of the ACM SIGCOMM 2009 conference on Data communication*, ser. SIGCOMM '09. New York, NY, USA: ACM, 2009, pp. 135–146. [Online]. Available: http://doi.acm.org/10.1145/1592568.1592585

[8] J. Finnis, N. Saigal, A. Iamnitchi, and J. Ligatti, "A location-based policy-specification language for mobile devices," *Pervasive and Mobile Computing*, vol. 8, pp. 402–414, 2010.

[9] B. Krishnamurthy, P. Gill, and M. Arlitt, "A few chirps about twitter," in *Proceedings of the first workshop on Online social networks*, ser. WOSN '08. New York, NY, USA: ACM, 2008, pp. 19–24. [Online]. Available: http://doi.acm.org/10.1145/1397735.1397741

[10] R. Gross and A. Acquisti, "Information revelation and privacy in online social networks," in *Proceedings of the 2005 ACM workshop on Privacy in the electronic society*, ser. WPES '05. New York, NY, USA: ACM, 2005, pp. 71–80. [Online]. Available: http://doi.acm.org/10.1145/1102199.1102214

[11] Nissenbaum, "Privacy as contextual integrity," *Washington Law Review*, vol. 79, no. 1, pp. 119–158, 2004.

[12] A. Barth, A. Datta, J. Mitchell, and H. Nissenbaum, "Privacy and contextual integrity: framework and applications," in *IEEE Symposium on Security and Privacy*, 2006, pp. 184–198.

[13] A. Iamnitchi, J. Blackburn, and N. Kourtellis, "The social hourglass: An infrastructure for socially aware applications and services," *IEEE Internet Computing*, vol. 16, pp. 13–23, 2012.

[14] A. K. Dey, G. D. Abowd, and D. Salber, "A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications," *Hum.-Comput. Interact.*, vol. 16, no. 2, pp. 97–166, Dec. 2001.

[15] OpenGroup, "Taking interoperability beyond the boundaries," http://www.opengroup.org/subjectareas/si, 2012.

[16] S. Kruk, "Foaf-realm: control your friends access to the resource," in *In Proceedings of the 1st Workshop on Friend of a Friend*, 2004.

[17] H. C. Choi, S. R. Kruk, S. Grzonkowski, K. Stankiewicz, B. Davis, and J. Breslin, "Trust models for community aware identity management," in *Proceedings of the Identity, Reference and Web Workshop, in conjunction with WWW 2006*, 2006, p. 140154.

[18] B. Carminati, E. Ferrari, and A. Perego, "Rule-based access control for social networks," in *Proceedings of the 2006 international conference on On the Move to Meaningful Internet Systems*, 2006, pp. 1734–1744.

[19] N. Elahi, M. Chowdhury, and J. Noll, "Semantic access control in web based communities," in *Proceedings of the 2008 The Third International Multi-Conference on Computing in the GlobalInformation Technology*, 27 2008-aug. 1 2008, pp. 131 –136.

[20] B. Carminati, E. Ferrari, R. Heatherly, M. Kantarcioglu, and B. Thuraisingham, "A semantic web based framework for social network access control," in *Proceedings of the 14th ACM symposium on Access control models and technologies*, ser. SACMAT '09, 2009.

[21] A. Masoumzadeh and J. Joshi, "Ontology-based access control for social network systems." *IJIPSI*, vol. 1, no. 1, pp. 59–78, 2011.

[22] R. Engelmore, Ed., *Readings from the AI magazine*. Menlo Park, CA, USA: American Association for Artificial Intelligence, 1988.

[23] P. W. Fong, "Relationship-based access control: protection model and policy language," in *Proceedings of the first ACM conference on Data and application security and privacy*, 2011, pp. 191–202.

[24] F. Giunchiglia, R. Zhang, and B. Crispo, "Relbac: Relation based access control," in *Fourth International Conference on Semantics, Knowledge and Grid*, 2008, pp. 3 –11.

[25] A. Squicciarini, F. Paci, and S. Sundareswaran, "Prima: an effective privacy protection mechanism for social networks," in *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security*, 2010, pp. 320–323.