Realizing Privacy by Definition in Social Networks

Matt Tierney
Google Inc. & NYU

Lakshminarayanan Subramanian
New York University (NYU)

Abstract

Privacy violations in online social networks (OSNs) have become more the norm than the exception. Conventional models of privacy in OSNs offer a limited set of privacy guarantees for information posted and reshared by users in OSNs. In this paper, we propose a completely new model of private information sharing using a refined abstraction of *contexts* that embodies the philosophy of *contextual integrity* (CI), which we believe better captures users privacy expectations in OSNs. We present the design of Compass, an online social network inspired by CI, in which three properties hold: (a) users are associated with *roles* in specific *contexts*; (b) every piece of information posted by a user is associated with a specific *context*; (c) *norms* defined on roles and attributes of posts in a context govern how information is shared across users within that context.

1 Introduction

Privacy and sharing are at odds in online social networks [14, 15, 36]. The current privacy setting models in OSNs have three basic design flaws: (a) there is often a mismatch between user-specified settings and the user perceived sharing intents; (b) those models offer inadequate privacy protection to the users; and (c) the systems upon which they are built do not verify the user's intentions. Johnson et al. [17] and Kairam et al [18] demonstrate the disconnect between Facebook and Google+ privacy controls and the settings that users believe they have set for their accounts [17]. Due to poor engineering practices, Facebook deployed flawed code that resulted in their CEO's private photos being leaked due to a bug in one of the "reporting flows" on the site [10]. In several court cases, users expectations of privacy have been unmet leading to the citation of evidence gathered on Facebook for a growing number of divorce proceedings [7, 33]. Even in casual settings, users have been unable to concurrently share their revelry with friends as well as maintain the standards expected of them by employers when representing themselves online [23].

We argue for a completely new model for how to think about privacy in OSNs. We contend that the right way to think about privacy is through the lens of *contextual integrity* (CI) [28], which provides concepts that more precisely describe how people conceive of privacy in the real world [26, 27] and therefore should guide how we design OSNs. Rather than focus on control and restriction, CI promotes an overarching idea of privacy as *appropriate flows of information*, the details of which have been applied to environments where privacy settings are well-understood, imperative, and nuanced [3, 29]. We argue that we can begin to apply CI to less codified privacy contexts; in particular, we can apply contextual integrity to OSNs.

In this paper, we propose Compass, a new social network design that is built with CI as its privacy core. The core idea of Compass is to create a universe of *contexts* where each context is reflective of privacy norms and practices in a specific real-world context. In Compass, a small collection of users (privacy experts) act as "administrators" to create new context definitions, which are associated with a set of roles and norms. New context definitions that are meant to be public need to be vetted by Compass context creators before being publicly posted; users can create private context definitions in Compass which are not public. In the common use case, Compass allows users to search for appropriate context def-

APSys '14, June 25-26, 2014, Beijing, China

initions (publicly defined) and create context instances which they control by assigning roles to users who subscribe to that context instance. By decoupling the definition of new contexts from most users, Compass aims to significantly simplify the actions required of a normal user of a system. A normal user who joins the system needs to only choose their roles within context instances of which they are a member. In other words, users join contexts for which norms are given – users do not modify information flows. Any information posted by a user is associated with a specific context and the norms of the context govern how the information can be shared within the context. By definition, information flow is restricted within each instance of a context. Users may belong to multiple contexts but, more importantly, information in one context cannot flow to other contexts. Compass transforms norms from human-readable logic [3] to a satisfiability problem for which we can (1) efficiently generate the binary decision diagram (BDD) for a norm set, (2) check adherence to norms, and (3) determine how to push a new post to a context in accordance with the norms. Compass compiles sets of context-sensitive informational norms (written in propositional logic) to access controls.

The design of Compass is primarily tailored to handle the case of privacy violations due to inadvertent leaks or inconsistent privacy policies which may lead to privacy violations. If an adversary launches a copy-and-paste attack, the primary and simple first-order defense offered by Compass is to perform similarity matching to detect potential norm violations (or "surprises") for original posters. A surprise notice enables users to signal to Compass regarding the appropriateness of the information flow. However, Compass is not explicitly designed to handle intentional adversaries who can arbitrarily modify content to violate privacy norms and evade detection.

2 Closely Related Work

New systems have emerged to enable users to think differently about their privacy in OSNs [4, 30]. Several projects influence, compare, and contrast with the design of Compass. Recently, Barkhuus examined the application of CI to the considerations of privacy in HCI works [2]. Aegis [19] takes a semantic web approach to writing policies for social networks. Compass proposes a new direction as it connects formal methods approaches to privacy [16] with a tangible implementation. Lipford, et al. [22] argue from the perspective of contextual integrity for how social networks ought to be designed for users' privacy. Guha, et al. [14] argue for a system that shares information in plain sight where users share data through publicly known dictionaries and privately shared seeds for the substitution cipher in the system. Anwar, et al. [13] illustrates how to articulate the Facebook access controls in an access control model.

3 Contextual Integrity Review

Nissenbaum proposed CI as a philosophical framework for understanding privacy [28]. *Contexts* are "structured social settings" [28] in which all other concepts operate. Contexts are the social spheres in which the framework of contextual integrity is applied. *Actors* are the entities (usually, people or organizations) that are a part of the context between which information flows. Actors are thus the individuals for whom the question of privacy is applied. Each of the actors takes on a *role* within a context. The role is simply a label that enables additional concepts in the context to compute appropriate flows of information. The concept of roles is similar to the concept of human-readable labeling for privacy purposes. However, within Compass, the value of roles is not simply to delineate access orderings [25] but to enable security principles that align with the aim of Compass: least privilege, separation of duties, and data abstraction [31]. The set of rules that describe how information flows through the context are the context-sensitive informational norms (or, simply, *norms*) of the context. Within CI, norms are not individually-designed rules or "settings" that determine how information should flow. Instead, norms are agreed upon sets of rules that guide how information ought to flow. This is key to understanding the CI perspective of norms: norms are given to the context by its designer. The CI framework describes how the determination of norms is a function of the actors (transmitter and, optionally specified, receiver(s)), roles of the actors, the subject (information to be transmitted), and the transmission principles.

4 Designing a CI-Aware OSN

Within Compass, a context begins as a user-created entity. Context definition creators have two basic actions for configuring a context: defining roles and choosing a norm set that governs information flows within roles. Compass has two types of context definitions: public contexts and private contexts. Compass promotes a small set of users, who are privacy experts, to articulate context definitions that match with standard privacy expectations with real-world privacy norms. Individual users can create their own context definitions which are considered private for their own consumption which they manage. Public context definitions are searchable by normal users while private context definitions are not exposed to other users. Private context definitions if more generally appli-

cable can be vetted by privacy experts in Compass before being made public (if the context creator wishes to make it public). To define a public context, a privacy expert user (approved by Compass) articulates a context definition for standard privacy practices in a real-world setting (similar to HIPAA) such as family settings, school, office etc. While certain areas like health care might have well-defined privacy policies, similar real-world scenarios (such as an office setting) may promote different context definitions for the same setting.

A normal user can search for publicly available context definitions and can create an *instance* of a publicly defined context. In the event of multiple context definitions for the same setting, the onus is on the user (creating the context instance) to choose the appropriate context definition for their setting. Any user who creates an instance of a context becomes the "administrator" for the context instance. Private context definitions automatically come with a context instance that the creator manages. An administrator of a context instance has three important actions: adding users, assigning roles to users , and acting on privacy violations (if a user acts in an adversarial manner and is discovered). Compass aims to provide isolation to each context instance; an information posted in one context instance cannot be shared to another context instance. Contexts and norms are preconfigured structures within Compass. By avoiding user-customized settings, we ensure that installed norms can be reasoned about and understood by users and experts who work on the Compass back end. The administrator invites other users to that context and if so begins to apply the other characteristics of the context. Users who join the context are assigned roles by the administrator of the context which defines the flow of information with the context through the norms.

The actions required from a normal user are made extremely simple in Compass: they receive invitations to participate with specific roles in different contexts and each user can choose to accept or reject such invitations. This is very similar to joining mailing lists or groups. Once they are part of a collection of contexts with specific roles, they can post information, receive posts from other users in the context or repost to users in the context.

4.1 Posts

Compass allows three actions that users may enact on posts: create, reply, and re-share. A user who creates a post provides the requisite fields (Table 1) through the Compass interface. A user who is a member of a context may post information to that context. Depending on the type of the information, the transmitter, and the declared receivers that information may be pushed to certain individuals in the context, searchable by individuals in the context, or entirely inaccessible to other individuals directly through their account in the context.

Parameter	Definition
transmitter	The actor who is sharing the information.
context	The social sphere in which the post is initially contained.
receiver	The individual(s), role(s), or entire context to which the transmitter to which the transmitter intends to push their post.
attributes	Any additional information that the norms may use to compute (1) to whom the post is pushed, (2) who can search for the post, and (3) who cannot access the post. Compass post attributes are either *context-sensitive* or *context-free*.

Table 1: Required elements of a post.

The user may create a post, indicating who should be the recipient of the information (an individual, role(s), or the entire context) as well as any relevant attributes. The context's norms evaluate how the information should flow. We have two classes of attributes: *context-free* and *context-sensitive*. Context-free attributes belong to publicly defined privacy classes (e.g., contains personally-identifiable information, contains objectionable material). For context-free attributes we have "generic" functions that operate on the message content, regardless of context. These generic rules are independent of the context-specific rules and do not affect the BDD construction or performance. A context-sensitive attribute is operates on a post within a specific context.

Compass programmers must translate these attributes to boolean return value functions given the context and the post as arguments. Replying to a post results in that reply and the original post only being visible according to the norms of the context.

4.2 Existing OSN Models in the CI Framework

The norms for Facebook can be represented as:

$$\text{inrole}(p_2, friend) \tag{1}$$

$$\text{inrole}(p_2, friend\text{-}of\text{-}friend) \tag{2}$$

The simplicity of the norms as they have been written suggest that simplicity does not guarantee flows of information [17]. Given that the role of friend is applied uniformly across all friends of an actor in the Facebook network, the permissiveness of this norm (Norm 1) and the Friends of Friends norm (Norm 2) are self-evident.

That is, the capacity to over-share or experience inappropriate flows of information is high. Similarly, the norms for a context as defined by a Google+ circle can be represented as:

$$\text{inrole}(p_1, \textit{circle-creator}) \wedge \text{inrole}(p_2, \textit{circle-member}) \quad (3)$$

$$\text{inrole}(p_1, \textit{circle-member}) \wedge (t \in \textit{limited}) \quad (4)$$

Norm 3 illustrates how sharing within a user-generated circle exists. What can be problematic about sharing and privacy on Google+ is what we see in Norm 4. In Norm 4 we see that a circle member can easily re-share information to users outside of a circle. Google+ does not protect the original posters from the inappropriate flows of information, making the friction to share with users outside of circle context nearly absent. A message indicating that the post was originally a limited share presented to the user, but the extent to which the original poster desires the post to remain within the circle is *not clearly indicated by the context*: no agreed upon norms exist to guide the appropriateness of the flow of information. Consequently, the re-sharer may inadvertently violate the expectations of the user. This is particularly problematic given that the original poster has no direct means of accessing with whom the res-hared post was shared: the post has escaped the isolation of the context.

4.3 Example Contexts in Compass

Here, we articulate three specific examples of norms for three different contexts and use these examples in our evaluation.

4.3.1 Family

Many types of family dynamics exist. For the purposes of our examples we focus on a small set of norms to discuss how family members interact in a social network context. We introduce assumptions about norms that we do not assume to be universally applicable; instead, we expect that when the reader will understand the conversion that we apply from the read norms to the propositional logic. Consider a family structure where family members (actors) have at least one of the following roles: *elder*, *generation*-0 (think: parent), and *generation*-1 (child). Moreover, we have attributes attached to messages that are sent through the context.

We assume that families contain a subset of members who are considered the mature, wise elders who make decisions about difficult topics. In Norm 5, we illustrate the use of the elder role to constrain the flow of information amongst users who have the role of elder when a message contains information about a genetic disease in the family. In the example family context, communication about a genetic disease of a parent is constrained to only the elders of family; e.g., information about Huntington's disease will only be shared with and amongst elders. Norms can also be topic specific for a generation. For instance, communication about finances remains between parents as in Norm 6. However, sharing about the children's low academic performance (Norm 7) or throwing parties (Norm 8) remains within their generation.

Notably, this example is imperfect. It may not be expressive of the family norms to which some readers are accustomed. The purpose of this subsection has been to demonstrate the means of expressing norms that convey a family dynamic. Different families, cultures, etc. will have different norms.

4.3.2 Classroom

We consider the case of a classroom with students that are divided into teams and an instructor. Students thus have both the role of *student* and as the member of a specific team, which we generically express as *team-member* in Figure 2. We examine a set of norms that govern the flow of information in various scenarios that affect the classroom dynamic between these roles and the individuals in the classroom. One norm is an instructor broadcasting announcements to the class. We see this permissive norm as Norm 9. Of course, if the instructor has a specific message pertaining to a specific team in the class, we see that there is a norm to ensure only members of that team receive that message in Norm 10. Undoubtedly, team members will want to be able to communicate amongst only themselves too (Norm 11). Questions about one's own grade are only received by the instructor (Norm 12). But gossip about a teacher is only seen by students in the class and not the instructor (Norm 13).

4.3.3 University Department

A university department may have a number of nuanced roles: administrators (seniors and aides), professors (tenured, tenure-track, non-tenure-track), students, and staff. On a sensitive matter regarding a students' disciplinary matter, an instructor may send a message that only the administrative board (tenured faculty who hear cases about students) will see (Norm 14). Communication from the chair about the tenure promotional process can only accessible to the tenured members of the department and the administrators (the *faculty-tenure-committee*). When professors are reporting student grades (Norm 16) or students reporting course ratings (Norm 17) only the administrators of the department see those scores, so as to act as mediators of initially sensitive information.

$$\text{inrole}(p_1, \textit{generation-0}) \wedge \text{inrole}(p_2, \textit{elder}) \wedge (q = p_1) \wedge (t \in \textit{genetic-disease}) \quad (5)$$
$$\text{inrole}(p_1, \textit{generation-0}) \wedge \text{inrole}(p_2, \textit{generation-0}) \wedge (t \in \textit{finances}) \quad (6)$$
$$\text{inrole}(p_1, \textit{generation-1}) \wedge \text{inrole}(p_2, \textit{generation-1}) \wedge (t \in \textit{low-academic-performance}) \quad (7)$$
$$\text{inrole}(p_1, \textit{generation-1}) \wedge \text{inrole}(p_2, \textit{generation-1}) \wedge (t \in \textit{parties}) \quad (8)$$

Figure 1: Norms of transmission for a family context.

$$\text{inrole}(p_1, \textit{instructor}) \wedge \text{inrole}(p_2, \textit{student}) \wedge (t \in \textit{announcement}) \quad (9)$$
$$\text{inrole}(p_1, \textit{instructor}) \wedge \text{inrole}(p_2, \textit{teamX-member}) \wedge (t \in \textit{teamX}) \quad (10)$$
$$\text{inrole}(p_1, \textit{teamX-member}) \wedge \text{inrole}(p_2, \textit{teamX-member}) \wedge (t \in \textit{teamX}) \quad (11)$$
$$\text{inrole}(p_1, \textit{student}) \wedge \text{inrole}(p_2, \textit{instructor}) \wedge (q = p_1) \wedge (t \in \textit{grade}) \quad (12)$$
$$\text{inrole}(p_1, \textit{student}) \wedge \text{inrole}(p_2, \textit{student}) \wedge (t \in \textit{instructor}) \quad (13)$$

Figure 2: Norms of transmission for a classroom context.

5 Translating Logic to Code

In order to evaluate human-readable norms as access controls, we implement a norm compiler that translates norms to a satisfiability problem data structure that can be queried regarding access controls. The norms, written as propositional logic, are compiled into *binary decision diagrams (BDD)* with additional generated code to represent a context. Given a post by a user in a context, a query to the BDD returns (1) if the message can be transmitted to users or roles in the context and (2) to which users or roles the message should be sent. There are two key components to the norm compilation: (1) connecting the grammar symbols to boolean return value functions and (2) generating the BDD.

From Grammar Symbols to Functions: We adopt the propositional logic grammar from Barth, et al. [3] and translate each of the propositional logic variables into functions with boolean return values. Across the various contexts defined, inrole() function calls are translated to database queries about the receiver and whether the message is appropriate for their access. When the post is posted, traversal of a p_2 node checks if the explicitly stated receivers meet the inrole() criteria to determine if a message push to their feed is appropriate. Alternatively, if a user requests direct access to a post, traversal of a p_2 node checks if the requestor's inrole() state is true or false for that node. $(t \in \ldots)$ functions are translated to attribute (attr()) function calls in our code. These functions exist as either explicit database checks (see if a post has a particular flag set for the attribute) or inferred base on the message content (e.g., vulgarity detection [35]). These functions directly query or operate on the post. $(q = \ldots)$ variables are translated to subject() function calls. These calls check that the subject of a message is the actor or role specified in the message. Depending on the context, the `attr()` are customized to either execute explicit field checks in a database or infer whether an attribute is present based on the content of the post.

Generating the BDD: A Binary Decision Diagram (BDD) is a directed, acyclic graph (DAG) representation of a boolean expression. In our example norm sets, all individual norms are represented as conjunctions. To determine whether to accept a post to a context and to whom to post it, we assume that the set of norms when considered in disjunction ("or"-ed together) accurately represents the intentions for information flows. Our lexer and parser produce a simple propositional logic AST that maintains the relationship between propositional logic operators ("and", "or", "negation") as well as method call state (function name and arguments).

After the norms are parsed by the compiler, the AST is analyzed to translate every node into a boolean variable, in order to map the function object to a boolean variable. Notably, we keep track of function signatures so as to not over generate boolean variables in the BDD analysis. We use BDDs for representing the problem of satisfying norms within the CI framework. We use a BDD library (in our case, BuDDy [21]) to input variables and propositional logic as a satisfiability problem. More importantly, we use a BDD library to heuristically produce a reordered and more-reduced BDD; in particular, a BDD that has many isomorphisms in its subgraphs merged. With a DAG representation of the BDD, for a norm set, we maintain a set of vectors of ordered transitions between nodes that represent the satisfiable paths of the DAG. (Each vector in the set represents a satisfiable path in the BDD.)

Evaluating Norms and Posts with BDDs: When considering to whom to push new posts, we assume that the traversal of nodes in a BDD (eventually, to a satisfying node) that contain a receiver (p_2) represent the list of valid receivers for a message. In the case that a receiver is somehow missed due to a short circuiting of a norm (for e.g., at least two norms are satisfied by the input, but only one triggers a push), users can currently poll for access to data. Given their state as a receiver, the BDD is traversed querying whether the receiver inrole() function

$$\text{inrole}(p_1, \textit{instructor}) \wedge \text{inrole}(p_2, \textit{administrative-board}) \wedge (q = \textit{student}) \wedge (t \in \textit{disciplinary-matter}) \quad (14)$$

$$\text{inrole}(p_1, \textit{chair}) \wedge \text{inrole}(p_2, \textit{faculty-tenure-committee}) \wedge (q = \textit{untenured-faculty}) \wedge (t \in \textit{tenure-case}) \quad (15)$$

$$\text{inrole}(p_1, \textit{instructor}) \wedge \text{inrole}(p_2, \textit{admin}) \wedge (q = \textit{student-grade}) \wedge (t \in \textit{grades}) \quad (16)$$

$$\text{inrole}(p_1, \textit{student}) \wedge \text{inrole}(p_2, \textit{admin}) \wedge (q = \textit{instructor}) \wedge (t \in \textit{course-rating}) \quad (17)$$

Figure 3: Norms of transmission for a university department context.

calls match the requestor's profile.

Handling Norm Updates: Context definitions can be modified by privacy experts over time and Compass enables constant checking for coherence of context norms over time. This is enabled by comparing the satisfiability sets of two different norms. Satisfying paths that do not overlap must be checked against one another to ensure that no bad informational flow surprises have been admitted to the norm set. We say that the system has *extensible verifiability* so that as the contexts evolve – requiring additional roles, nuanced norms, etc. – the framework allows the programmers to make clear what past norms may be violated so as to limit inducing privacy surprises. Any differences, especially in terms of additional satisfying paths, must be inspected to determine if a conflict has significant social impact.

6 Surprise Information Flows

The basic design of Compass is designed around the principle of contextual integrity and is not designed to handle adversaries whose explicit goal is to violate the privacy norms. Adversarial users may violate the norms of the original poster by launching a copy-paste attack where they explicitly "copy" information received in one context and post it as new information to users in a different context. Compass explicitly disallows such reposting and adversaries need to perform a copy-paste operation to achieve this goal. To detect copy-and-paste (C&P) attacks, we rely on similarity measures for text [5, 24] (we plan to use the library facilities of NLTK [6]), natural images [34], and video [12, 32]. Whenever a new post of a user to a context is deemed very similar to a post received by the same user from a different context, we detect a potential copy-paste attack. We refer to such a case as a *surprise information flow*. If the information appears to be an "exact" copy of the original, then such a post is disallowed and the original poster is alerted when this norm-contrary surprise occurs. If the information appears to be approximately similar, then the user posting the information is requested to attest the originality of the posting and the original poster is informed of a potential surprise without revealing the new information posted; in this case, the original poster in doubt can request the user posting the new information to either share the posted information or request a similarity report from Compass.

For any surprise information flow, Compass informs the original poster of the retransmitter as well as the context with which the information was shared. When a violation occurs, users may find multiple reasons to take issue with the re-sharing that occurred. The parameters of a post are extended to include those that the user may find are appropriate to classify the privacy violation: time, new context, new users, the content, etc. Ideally, one would require a reputation system on top of Compass to resolve conflicts across original posters and potential reposters in the case of multiple conflicting claims about information originality for similar-looking posts. For any declared privacy violation, Compass logs the user's feedback about whether the surprise was acceptable or not and why. The user may then choose to decrement the reputation of the individual who inappropriately re-shared the information or the context to which it was re-shared. This scoring, relative to the user as well as globally maintained, enables all users to assess the credibility of individuals within the Compass ecosystem. Administrators of contexts have the power to remove users with low reputation (involved in several conflicts) from a context.

7 Implementation and Status

We have implemented the core functionalities of the Compass design in order to demonstrate concretely how to realize contextual norms of transmission as code and evaluate the complexity of the norms constructs in example contexts. The code has been implemented in about 1000 lines of C++, flex, and bison as libraries that utilize BuDDy [21] for binary decision diagrams (BDDs) [1], thrift [11] for data structure serialization, leveldb [8] for the key-value stores, as well as bison [9] and flex [20] for the core compiler components. To generate the BDD, the compiler executes two passes over the norms: once to gather the *variables* for the propositional logic and the second to generate the BDD once all variables have been determined. Our preliminary evaluation based on the example contexts we described in Section 4.3 and formally in Section 5 show that (a) Our Compass compiler can easily translate real-world context definitions to generate BDDs for different contexts (b) The verification code generated by our naive implementation for verifying information flow for a context is highly efficient and can verify the correctness of information flow for 10,000 queries per second. We are currently implementing a

complete social network based on the Compass privacy model.

References

[1] AKERS, S. B. Binary decision diagrams. *Computers, IEEE Transactions on 100*, 6 (1978), 509–516.

[2] BARKHUUS, L. The mismeasurement of privacy: using contextual integrity to reconsider privacy in hci. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems* (2012), ACM, pp. 367–376.

[3] BARTH, A., DATTA, A., MITCHELL, J. C., AND NISSENBAUM, H. Privacy and contextual integrity: framework and applications. In *2006 IEEE Symposium on Security and Privacy (S&P'06)*.

[4] BIELENBERG, A., HELM, L., GENTILUCCI, A., STEFANESCU, D., AND ZHANG, H. The Growth of Diaspora - A Decentralized Online Social Network in the Wild. In *IEEE INFOCOM 2012 - IEEE Conference on Computer Communications Workshops*.

[5] BILENKO, M., AND MOONEY, R. J. Adaptive duplicate detection using learnable string similarity measures. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining* (2003), ACM, pp. 39–48.

[6] BIRD, S. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions* (2006), Association for Computational Linguistics, pp. 69–72.

[7] CLAYTON, R. B., NAGURNEY, A., AND SMITH, J. R. Cheating, breakup, and divorce: Is facebook use to blame? *Cyberpsychology, Behavior, and Social Networking* (2013).

[8] DEAN, J., AND GHEMAWAT, S. LevelDB. `https://code.google.com/p/leveldb`.

[9] DONNELLY, C., AND STALLMAN, R. M. *Bison: The YACC-compatible Parser Generator (November 1995, Bison Version 1.25)*. Free Software Foundation, 1998.

[10] DUELL, M. Mark Zuckerberg's private Facebook photos revealed: Security 'glitch' allows web expert to access billionaire's personal pictures. *The Daily Mail (MailOnline)* (December 2011).

[11] FACEBOOK. Apache Thrift. `http://thrift.apache.org`.

[12] FLICKNER, M., SAWHNEY, H., NIBLACK, W., ASHLEY, J., HUANG, Q., DOM, B., GORKANI, M., HAFNER, J., LEE, D., PETKOVIC, D., ET AL. Query by image and video content: The qbic system. *Computer 28*, 9 (1995), 23–32.

[13] FONG, P. W., ANWAR, M., AND ZHAO, Z. A privacy preservation model for facebook-style social network systems. In *Computer Security–ESORICS 2009*. Springer, 2009, pp. 303–320.

[14] GUHA, S., TANG, K., AND FRANCIS, P. Noyb: Privacy in online social networks. In *Proceedings of the first workshop on Online social networks* (2008), ACM, pp. 49–54.

[15] HENNE, B., SZONGOTT, C., AND SMITH, M. Snapme if you can: privacy threats of other peoples' geo-tagged media and what we can do about it. In *Proceedings of the sixth ACM conference on Security and privacy in wireless and mobile networks* (New York, NY, USA, 2013), WiSec '13, ACM, pp. 95–106.

[16] JAYARAMAN, K., GANESH, V., TRIPUNITARA, M., RINARD, M., AND CHAPIN, S. Automatic error finding in access-control policies. In *Proceedings of the 18th ACM conference on Computer and communications security* (New York, NY, USA, 2011), CCS '11, ACM, pp. 163–174.

[17] JOHNSON, M., EGELMAN, S., AND BELLOVIN, S. M. Facebook and privacy: it's complicated. In *SOUPS '12: Proceedings of the Eighth Symposium on Usable Privacy and Security* (2012).

[18] KAIRAM, S., BRZOZOWSKI, M., HUFFAKER, D., AND CHI, E. Talking in circles: selective sharing in google+. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2012), CHI '12, ACM, pp. 1065–1074.

[19] KAYES, I., AND IAMNITCHI, A. Aegis: A semantic implementation of privacy as contextual integrity in social ecosystems. In *11th International Conference on Privacy, Security and Trust (PST)* (July 2013).

[20] LEVINE, J. *Flex & bison*. O'Reilly Media, 2009.

[21] LIND-NIELSEN, J. Buddy bdd library, 2002.

[22] LIPFORD, H. R., HULL, G., LATULIPE, C., BESMER, A., AND WATSON, J. Visible flows: Contextual integrity and the design of privacy mechanisms on social network sites. In *Computational Science and Engineering, 2009. CSE'09. International Conference on* (2009), vol. 4, IEEE, pp. 985–989.

[23] LOVE, D. 17 People Who Were Fired For Using Facebook.

[24] METZLER, D., DUMAIS, S., AND MEEK, C. Similarity measures for short segments of text. In *Advances in Information Retrieval*. Springer, 2007, pp. 16–27.

[25] MYERS, A. C., AND LISKOV, B. Protecting privacy using the decentralized label model. *ACM Transactions on Software Engineering and Methodology (TOSEM) 9*, 4 (2000), 410–442.

[26] NARAYANAN, A. What happened to the crypto dream?, part 1. *Security & Privacy, IEEE 11*, 2 (2013), 75–76.

[27] NARAYANAN, A. What happened to the crypto dream?, part 2. *IEEE Security & Privacy 11*, 3 (2013), 0068–71.

[28] NISSENBAUM, H. *Privacy in context: Technology, policy, and the integrity of social life*. Stanford Law Books, 2009.

[29] NISSENBAUM, H. A contextual approach to privacy online. *Daedalus 140*, 4 (2011), 32–48.

[30] PANG, J., AND ZHANG, Y. A new access control scheme for facebook-style social networks. *arXiv preprint arXiv:1304.2504* (2013).

[31] SANDHU, R. S., COYNE, E. J., FEINSTEIN, H. L., AND YOUMAN, C. E. Role-based access control models. *Computer 29*, 2 (1996), 38–47.

[32] SANTINI, S., AND JAIN, R. Similarity measures. *Pattern analysis and machine intelligence, IEEE transactions on 21*, 9 (1999), 871–883.

[33] STEVENS, J. The Facebook divorces: Social network site is cited in 'a THIRD of splits'.

[34] WANG, Z., BOVIK, A. C., SHEIKH, H. R., AND SIMONCELLI, E. P. Image quality assessment: From error visibility to structural similarity. *Image Processing, IEEE Transactions on 13*, 4 (2004), 600–612.

[35] XIANG, G., FAN, B., WANG, L., HONG, J., AND ROSE, C. Detecting offensive tweets via topical feature discovery over a large scale twitter corpus. In *Proceedings of the 21st ACM international conference on Information and knowledge management* (2012), ACM, pp. 1980–1984.

[36] ZHANG, C., SUN, J., ZHU, X., AND FANG, Y. Privacy and security for online social networks: challenges and opportunities. *Network, IEEE 24*, 4 (2010), 13–18.