Privacy architectures in social networks state-of-the-art survey

Mohammad Badiul Islam*

Software Systems Research Group,
NICTA, 70-72 Bowen St., Spring Hill,
Brisbane QLD 4000, Australia
and
Queensland University of Technology (QUT),
2 George St, Brisbane QLD 4000, Australia
Email: mb.islam@qut.edu.au
*Corresponding author

Renato Iannella

KnowledgeFlux,
Level 7, 100 Edward St, Brisbane 4000, Australia
and
Queensland University of Technology,
2 George St, Brisbane QLD 4000, Australia
Email: r@iannel.la

Jason Watson

Science and Engineering Faculty,
Information Systems School,
Queensland University of Technology,
Gardens Point, Y Block Level 7, 706C,
CRICOS No. 00213J, Australia
Email: ja.watson@qut.edu.au

Shlomo Geva

Science and Engineering Faculty,
Electrical Engineering, Computer Science, Computational Intelligence and Signal Processing, Queensland University of Technology,
Gardens Point, S Block Level 10,
2 George St., Brisbane QLD 4000, Australia
Email: s.geva@qut.edu.au

Abstract: The use of social networking has exploded, with millions of people using various web- and mobile-based services around the world. This increase in social networking use has led to user anxiety related to privacy and the unauthorised exposure of personal information. Large-scale sharing in virtual spaces means that researchers, designers and developers now need to re-consider the issues and challenges of maintaining privacy when using social networking services. This paper provides a comprehensive survey of the current state-of-the-art privacy in social networks for both desktop and mobile uses and devices from various architectural vantage points. The survey will assist researchers and analysts in academia and industry to move towards mitigating many of the privacy issues in social networks.

Keywords: social networks; privacy; access control; distributed social networks; mobile social networks; open source social networks; Privacy by Design; PbD.

Reference to this paper should be made as follows: Islam, M.B., Iannella, R., Watson, J. and Geva, S. (xxxx) 'Privacy architectures in social networks state-of-the-art survey', *Int. J. Information Privacy, Security and Integrity*, Vol. X, No. Y, pp.000–000.

Biographical notes: Mohammad Badiul Islam is a researcher at National ICT Australia (NICTA). His research interests include non-classical logics (e.g., modal, deontic, defeasible logics), automated reasoning and their applications to normative reasoning, multi-agent systems, business processes, knowledge representation, privacy, social media and Web 2.0. He has more than ten years of research and technical work experience in Australia, Sweden and Bangladesh in various roles in various multinational organisations. His in-depth research, programming and web skills and knowledge are founded on his background in analysis, usability, design and development. This extensive skill-base along with his multinational experiences, helps ensure project success. He is also an experienced leader, team player, product manager and business analyst where his interdisciplinary and international skills have also added value. He is a former Brisbane International Student Ambassador for Bangladesh, which helped to create global advocates for Brisbane, Australia and raise cross-cultural awareness.

Renato Iannella is the Head of Innovation and Emerging Technologies at KnowledgeFlux and provides technology, architecture, and business insights into semantic data, visualisation, and collaboration. His experience covers enterprise information and data architecture, Semantic Web and linked data, social media, rights and privacy management, and data governance which he consults to industry and government. He is an active member and chair of numerous standards groups in web technologies and was a former member of the World Wide Web Consortium (W3C) Advisory Board. He holds adjunct professorship positions at both Queensland University of Technology and the University of Hong Kong. He was previously Lead Information and Policy Architect at NEHTA, Principal Researcher and Group Manager at the National ICT Australia (NICTA), Head Information Architect at IPR Systems, and Principal Scientist at the Distributed Systems Technology Centre (DSTC).

Jason Watson coordinates Digital Environments Education in the Science and Engineering Faculty at Queensland University of Technology, Australia. His research interests include understanding human behaviour on social platforms, specifically, how understanding behaviour informs social technology adoption and social technology systems design. He also leads a research team investigating how social technologies are reshaping contemporary

organisations and has been an active researcher and lecturer in both the UK and Australia. He is well published in the field of technology, education, and social technologies and successfully instigated and completed research projects in this these spaces.

Shlomo Geva leads Artificial Intelligence and Image Processing, Computer Software, Library and Information Studies at Queensland University of Technology, Australia. His research interests include web search, XML search engines, text search engines, link discovery, information retrieval and document computing.

1 Introduction

A social network (SN) is a website or network of connections and relationships. In general, it is defined as "a network of interactions of relationships" (Aggarwal, 2011), while its most classic definition is "the interactions of humans" (Aggarwal, 2011) in a platform. Some of the most recent and leading SNs are Facebook (Facebook Inc., 2015), LinkedIn (LinkedIn Corporation, 2011), Twitter (Twitter Inc., 2014), or content-sharing networks such as YouTube (YouTube LLC, 2014) and Flickr (Yahoo! Inc., 2004). In these (and other) SNs, a person seeks to discover like-minded or compatible people with interests or experiences similar to theirs (Chatterjee, 2013). Most recent SNs combine elements of various types of information, such as music, photos, videos, blogs, links, and third party applications.

The practice of social networking has exploded, with millions of users across many and various web-based services. SNs have become an important part of user social identity. People use their personal information to create a social profile, and then devote substantial time and energy to maintaining and manipulating their online persona in the SNs. Thus, SNs have transformed the web into a new medium for social communities to share personal data such as contacts, pictures, activities, and other personally identifiable information (PII). The intent of SNs is to facilitate connection and sharing, however, sharing personal data has consequences and SNs users and specialists are looking at the privacy impact of such large-scale sharing; for example, 90% of 5,627 respondents in 22 countries expressed anxiety about information privacy and ranked privacy issues as troubling and expressed anxiety about information privacy (KPMG International Cooperative, 2010).

There is a fundamental conflict between SN objectives and privacy protection. By definition, SNs promote sharing through SN functionalities; this sharing is the very purpose of their existence. Privacy is not considered evil; indeed, in some circumstances, any necessary protection may itself be considered evil by some SN users. The question is, therefore, how SN users can be empowered to define their own (flexible and changeable) privacy preferences, and how they might be assured that protection is suitably implemented to deliver these preferences through SN functionalities. The primary objective is, therefore, to understand and manage user privacy requirements and protection as a mechanism for delivering reliably managed privacy policies through SN functionalities.

The privacy problem is complex: users want privacy but they seldom know 'how to specify' and 'what to seek' for their own privacy (Shapiro, 2010). Embedded

privacy-enhancing technology (PET) (Blarkom et al., 2003); at the design level of SNs architectures can be the solution for ensuring privacy from the beginning of a system's development. PET is a coherent information and communication technology (ICT) system that protects privacy by eliminating or reducing unnecessary disclosure, collection, retention, sharing, or trading of personal data without losing the functionality of information systems.

This paper provides an overview of the current literature pertaining to privacy in SNs architectures. This area is of particular interest both in academia and in industry since SNs are so saturated. Hodge (2006) aptly points out that SNs are designed as public spaces for the private individual; however, since individuals use these public spaces to disseminate their personal information, the networks become the source of complex privacy issues. Additionally, user information should not be perpetually stored and accessed by the service provider. Stored information can be stolen or hacked, or a user account can be deactivated or reactivated at any later stage. It is likely that users would always wish for full ownership and control of their information so that they could turn off access at any time; however, in the current various SN paradigms, the owner of the information – service provider or user – remains unclear. This paper focuses on these SN privacy issues as documented by analysts, researchers, and users. In so doing, it determines the issues that need to be addressed in this area.

In the following sections, we begin by examining studies of privacy protecting architectures. We then describe the studies of architecture models. Next, we outline the representative results of privacy protection mechanisms in SNs. The following section describes the studies of enterprise architecture methodologies. We conclude with a discussion of implications for future research and practice.

2 Privacy protecting architectures

In information systems, architectures can be classified into six categories based on architecture's scope/objectives, business model, information system description, technology model detailed description and machine language description and each of these categories can be described for "*what (e.g. data), how (e.g. function), where (e.g. network), who (e.g. people), when (e.g. time) and why (e.g. motivation)*". Therefore, altogether there are thirty-six possible information system architectures (Zachman, 1987). These architectural separations liberate us to classify privacy protecting architectures from single architectural perspective and viewed architectures into various perspectives.

In the prior studies, architecture has been viewed 'strategically', 'organisationally' and 'technologically' (Iyer et al., 2007). This perspective, while influential individually, falls short to provide integrated view of architecture to analyse risk and decide upon resource allocation and this perspective often concentrate on idealised system instead of system in use (Iyer et al., 2007). Iyer and Gottlieb (2004) identified three views of the architectures: 'architecture-in-design (AID)'; 'emergent' and 'architecture-in-operation (AIO)'. The AID is also known as conceptual architecture (Iyer and Gottlieb, 2004) or 'espoused' (Iyer and Gottlieb, 2004), defines and models the architecture and describes the planned dependences between system modules. The emergent view is the actual dependencies that exist among system modules which allows to merge and acquiesce applications into the enterprise. The third view – AIO or 'architecture-in-use' (Iyer and Gottlieb, 2004) sketches the dependencies that arise from the business of doing the work

of the enterprise such as 'selling products', 'buying supplies', 'managing employees or suppliers' or managing other stakeholders interact with the system. The AID is used to define enterprise model and requisite organisational resources and the AIO provides the content of the enterprise's model. These views are also referred as sub-architectures and are of paramount importance in defining enterprise strategies (Iyer and Gottlieb, 2004). However, these sub-architectures may subsequently emerge from one to another and may be limited to provide a holistic view of the SNs architectures.

In the current SNs paradigm particularly in client-server architecture, users store their information in the public storage in SNs and access their information by using various access mechanisms in the virtual space owned by the service provider. A variety of methods is available to store the information; however, these are limited in their application. An improved system architecture for the storage of user information is now necessary; ideally, the SN provider would provide public storage for private user information, but would be unable to view or access this information without the person's explicit permission. So, examining privacy protecting architectures from a storing information standpoint might be useful since storing information is a significant challenge in SNs, particularly in terms of time link: information that was once public can be transformed to private information at a later stage, or private information can later be considered public information. In the client-server architecture SN paradigm, the service provider has the opportunity to search, view or access user information at any stage. While this is in line with the service provider's policy to which users have agreed, the service provider can revise their privacy policy and disclose personal information to a third party at any time.

There are five different system architectures for storing information in SNs this classification is inspired by the system architectures perspective by Chow and Mokbel (2009) for privacy-preserving location-based services. These are: client-server architecture (centralised architecture); third party architecture (similar to cloud-based architecture); distributed architecture (requiring fixed communication architecture, that is, a base station); peer-to-peer (P2P) or ubiquitous architecture (not requiring a fixed communication architecture); and wireless sensor networks (where sensor nodes provide aggregate information). Refer to Table 1 for architectures' summaries with projects and privacy features. While prior research has shown that all these system architectures have various pros and cons, a number of issues such as information privacy and reliability remain unsolved.

2.1 Client-server architecture

Client-server architecture is also known as 'centralised architecture' and uses a central repository to store data about their users and their connection (Bortoli et al., 2009). The SN service providers facilitate a set of services such as finding other people, sharing pictures/videos, and exchanging professional and personal information. Some renowned SN services such as Facebook (Facebook Inc., 2015), MySpace (Myspace LLC, 2014) and LinkedIn (LinkedIn Corporation, 2011) are developed based on this architecture. Existing work in this architecture can be divided into:

1 access control mechanism

2 virtual and individual client/server approach

3 user-centric approach.

With *the access control mechanism*, users may specify that certain of their profile items are accessible only by 'friends', 'friends of friends', or certain members in the friend list by using an access control list (Lugano and Saariluoma, 2007), multilevel access control list (Park et al., 2010), degree of relationship (Cai et al., 2009), or hidden friendship matching (Preibusch and Beresford, 2009). The users may iteratively capture their privacy preferences by using 'Privacy Wizard' (Fang and LeFevre, 2010) or may use an automated service such as 'Privacy Butler' (Wishart et al., 2010). However, in this approach, SN users are required to make a substantial effort and devote time to setting up access control for others so as to achieve partial privacy and protection from future privacy breaches.

Virtual and individual client/servers (Cáceres et al., 2009) or personal databases such as 'MyLifeBits' (Park et al., 2010), and 'Phonebookmark' (Ekler and Lukovszki, 2010) can be used to store user data or identity (Beach et al., 2009) in an individual server, rather than uploading the data to a centralised server. Such a server might resolve privacy issues such as direct anonymity issues, indirect or K-anonymity issues, eavesdropping, spoofing, replay, or wormhole attacks and allows for the use of a location-based system without disclosing user identity. However, one limitation is that the system uses of centralised server, which may highlight user future privacy issues.

In *the user-centric approach*, the user can manually or automatically set up their trustable mechanisms (VENETA; Von Arb et al., 2008), use a combined ontology for both class membership and family relationships (smart architecture, Noll et al., 2007). The user can also use a time capsule with timed and revocable decryptability to use SN services with anonymity and less trust in an external authority (Camenisch et al., 2009). However, this approach may not ensure overall service privacy, but may partially do so through a friend list or the selective sharing of information with service providers, or with friends, family and colleagues ('PeopleFinder'; Sadeh et al., 2009).

2.2 Distributed architecture

SN services are decentralised and distributed across various providers in distributed architecture. The SN facilitates services through widgets, plug-ins or add-ons to implement functionality on user websites. This architecture is also known as federated SN architecture. Existing work in this architecture can be divided into (GNU social, 2014b):

1 commodity web hosting

2 non-free software

3 federation of servers

4 P2P and distributed hash table (DHT)

5 social desktop applications

6 in-browser profile and certificates

7 distributed node architecture.

Refer to Section 4.3 for details description for these types of distributed architectures.

2.3 Third party architecture

Third party architecture uses third party storage; this is also known as 'anonymiser' and is placed between users and service provider as a middle layer. This layer must satisfy user privacy requirements (Chow and Mokbel, 2009) while storing information. In this architecture, user information can be transformed into blurred, cloaked, hidden, or encrypted information and stored in:

1 a trusted third party or broker [for example, 'SmokeScreen' (Cox et al., 2007)]

2 an untrusted third party (Puttaswamy and Zhao, 2010).

This transformed information will be unblurred, uncloaked, made visible or decrypted in client devices for further use.

2.4 P2P or ubiquitous architecture

In P2P or ubiquitous architecture, a user of a mobile or hand-held device can establish social networking by identifying another user who is both close by and using another mobile device (Chatterjee, 2013). This architecture generally uses overlay or logical networks. There is no fixed communication infrastructure or centralised/distributed servers; mobile users directly communicate to each other through the multi-hop routing P2P/DHT (GNU social, 2014b) approach. The users can engage in SN services such as sharing information, pictures, or locations. This approach might be one of the best for ensuring privacy as it allows any social interaction to be end-to-end or group encrypted. However, the P2P approach requires a special strategy for message delivery when a source server, DHT and/or group communication goes 'offline'. Some examples of this type of SNs approach are: PeerSoN (PeerSoN, 2016) and Opera Unite (Opera Software ASA, 2011).

2.5 Wireless sensor network architecture

Wireless sensor network architecture is based on wireless sensor data. Existing work in this type of SN takes two main directions:

1 dividing the system into hierarchical levels on physical units, for example, sub-rooms, rooms, floors, smart buildings or between cars

2 providing an in-network information anonymisation algorithm, regardless of the system's physical structure (Chow and Mokbel, 2009).

In this architecture, SNs can be used as 'storage infrastructures' for sensor information (Breslin et al., 2009). Users can also avail themselves of both the capabilities of the Semantic Web and mobile ad-hoc networks to ensure privacy based on a simple number as the trust mechanism – such as the 'BlueTrust' system (Markides and Coetzee, 2008) – or the friendship initiation system, 'serendipity', which is one of the initial Bluetooth-based systems (Eagle and Pentland, 2005). However, trust is more complex, and its regular occurrence in the Bluetooth range is not enough to create symmetric or asymmetric trust. Users can also initiate friendship by sharing their social networking ID, [e.g. WhoZThat (Beach et al., 2008)], extended to a context-aware component such as

Music Jukebox. However, this linkage to SN information may raise future privacy breaches.

MyLifeBits is a personal database-based system which is based on converging and collaborative computing (Park et al., 2010). 'Smart architecture' enables SN user privacy based on the identity of the user, using a combined ontology for both class membership and family relationships (Noll et al., 2007). 'Novel architecture' is an accountable privacy-supporting service which uses time capsule (Camenisch et al., 2009). These architectures partially or completely address privacy issues using their own built-in systems. However, they too have limitations; for example, MyLifeBits was not verified in a real life environment or smart architecture, and overall user privacy was partially substantiated through an access control mechanism for a SN's friend list, but was not confirmed overall.

End-to-end privacy protection should be integrated into the design stage of SN service development. Although many designers incorporate privacy protection techniques for a particular component of their SN, the literature reveals that for optimum safety, they need to be embedded at the design stage.

Table 1 Architectures' summaries with privacy features

Project name		*Features*	*Privacy support*
		Architecture type: client-server architecture	
1	Facebook (Facebook Inc., 2015)	Structure: news feed, friend, wall, timeline, like, messages, inbox notifications, networks and groups. Applications: events, marketplace, notes, places, platform, questions, photos, videos and Facebook paper. General features: credits, feature phones, graph search, IPv6, listen with friends, Facebook Live, mood faces, phone, poke, smart phone integration, subscribe, ticker, URL shortener, verified accounts, hash tagging feature, introducing say thanks, impressum and call to action button	Extensive but users are required to make a substantial effort and devote time for setting up privacy. Unclear data ownership and users grant a non-exclusive, transferable, sub-licensable, royalty-free, worldwide license to use any IP content that they post on or in connection with Facebook
2	MySpace (Myspace LLC, 2014)	Bulletin board, Groups, embed YouTube videos, instant messenger, MySpaceTV, mobile phone version, news, classifieds, applications, audio recordings, polls forum system and status update feature	Extensive but users are required to make a substantial effort and devote time for setting up privacy. Unclear data ownership and automatic data storing and collection by MySpace Services and their third-party service providers increase the risk of privacy issues

Note: *Some of the distributed architectures[1], features and privacy support are adapted from Wikimedia Foundation Inc. (2015).

Table 1 Architectures' summaries with privacy features (continued)

Project name		*Features*	*Privacy support*
		Architecture type: client-server architecture	
3	LinkedIn (LinkedIn Corporation, 2011)	User profile network, security and technology, applications, mobile, groups, job listings, online recruiting, skills, publishing platform, influencers, advertising and for-pay research	Extensive but users are required to make a substantial effort and devote time for setting up privacy. They may have access to user data which is a threat to user's privacy.
4	Twitter (Twitter Inc., 2014)	Tweets, content, format, trending topics, adding and following content, verified accounts, mobile, authentication, related headlines feature	Yes but the service provider have access to user data which is a threat to user's privacy.
5	Privacy Wizard (Fang and LeFevre, 2010)	Iteratively capture user's privacy preferences by using 'Privacy Wizard'	Yes through 'Privacy Wizard'
6	Privacy Butler (Wishart et al., 2010)	Use of an automated service	Yes through automated 'Privacy Butler' service
7	MyLifeBits (Park et al., 2010)	Use personal database	Yes through personal database
8	Phonebook mark (Ekler and Lukovszki, 2010)	Store user data or identity in an individual server, rather than uploading the data to a centralised server	Protect privacy by storing user data or identity in an individual server, rather than uploading the data to a centralised server
9	VENETA (Von Arb et al., 2008)	User can manually or automatically set up their trustable mechanisms	Yes through trustable mechanisms
10	Smart architecture (Noll et al., 2007)	Uses of a combined ontology for both class membership and family relationships	Provide privacy feature through combined ontology
11	Novel architecture (Camenisch et al., 2009)	Use time capsule with timed and revocable decryptability to use SN services with anonymity and less trust in an external authority	Yes
		*Architecture type: distributed architecture**	
1	6d[2]	Blog, media library, address book, themeable, private messaging	Address book to send posts to either individuals or groups.
2	Ampify[3]	Trust-based search	Provides fine grained privacy control through object capability security and transport layer encryption.
3	Anahita[4]	Anahita is an open source social networking platform for building knowledge sharing apps and services	

Note: *Some of the distributed architectures, features and privacy support are adapted from Wikimedia Foundation Inc. (2015).

Table 1 Architectures' summaries with privacy features (continued)

Project name		*Features*	*Privacy support*
		*Architecture type: distributed architecture**	
4	Appleseed[5]	Photos, journals, messaging, groups, privacy controls, status updates, newsfeeds	Friend circles used to categorise friends and restrict/allow access
5	Buddycloud[6]	Personal and topic channels, Buddycloud directory, channel search, channel recommender, media server, friend-finder, mobile and e-mail push service, location, messaging	Yes
6	Cunity[7]	Friends, photo album, filesharing, messaging, pinboard, news feed, member list, forum, connecting cunities	Yes
7	Diaspora*[8]	Status messages, blogging, photo sharing, privacy enhanced	Yes, through 'aspects'
8	DiSo Project[9]	Open, non-proprietary and interoperable decentralised social web	Not found
9	DSNP[10]	Provides the social web into an open space were everyone is free to contribute to an ecosystem of software and techniques	Through RSA asymmetric key
10	Duuit![11]	Search, micro-blogging, e-mail, photos, videos, blogs, web pages, XMPP chat, video chat, collaborative drawing, document creation and editing, feed reader, profiles, files, games, groups, mood, privacy controls, customisable interface	Excellent, acl, granular, profile, avatar, content
11	Friend2Friend[12]	Strong encryption, XML for all data exchange, data is digitally signed	Connect to known individuals.
12	Friendica[13]	Rich profiles, networking groups, community/group/celebrity pages, richtext status (not specifically length limited), photo albums, YouTube share, location, like/dislike, multiple profiles w/assignment to specific friends, single sign on to post directly to friend's profiles on co-operating systems. Communications encryption. Fans and one-way relationships. Local and global directory services. Ability to restrict connection endpoints.	Extensive
13	GNU Social[14]	Micro blogging	Yes but not clear 'how'
14	Jappix[15]	XMPP client + Micro blogging	Excellent: based on presence authorisations

Note: *Some of the distributed architectures, features and privacy support are adapted from Wikimedia Foundation Inc. (2015).

Table 1 Architectures' summaries with privacy features (continued)

Project name		*Features*	*Privacy support*
		*Architecture type: distributed architecture**	
15	Knowee[16]	OpenID signup, activity stream import and export, contact import from Web 2.0 services via XFN and FOAF, automatically updated address book from remote data sources, consolidated profile with RDF/FOAF export, personal SPARQL API	Yes but USER authorises KNOWEE to disclose the data supplied by the user by any means stated in them
16	Kopal[17]	OpenID Core, multiple profiles	Not found
17	Kune[18]	real-time collaborative edition, XMPP chat, groups, calendar, lists, tasks, blogs, Apache Wave inbox (modern e-mail), wave extensions (gadgets, robots), public web pages, profiles, galleries (photos, videos), maps, federation, usability	Excellent
18	Lipsync.it[19]	Synchronisation tool inspired by Dropbox	Yes
19	Libertree[20]	SN	No
20	Lorea[21 22]	Profiles, micro blogging, streams, groups, plugins, group mailing lists, tasks, calendar, subgroups, tag clouds	Excellent
21	Movim[23]	XMPP client + Micro blogging	Not yet
22	Mr. Privacy[24]		Yes
23	Newebe[25]	One user = one node; micro blogging, picture sharing, activity stream	
24	NoseRub[26]		Not found
25	ObjectCloud[27]	Customisation, flexible hosting, security, application platform	Yes
26	OneSocial Web[28]	Micro blogging	Yes
27	OpenAutonomy[29]	Micro-blogging, RSS aggregation, cloud storage	Trusted user list and fine-grained trusted sub-groups
28	OpenLink Data Spaces (ODS)[30]	Profile management, blogs, wikis, address books, calendars, feed aggregation, discussion forums (includes NNTP support), file servers (WebDAV-based briefcase).	WebID and others
29	OpenMicroBlogger[31]	User-toggleable 'apps' to add/remove functionality. RSSCloud and partial OStatus (PubSubHubbub) federation as well as Open Microblogging 0.1. Local follow/unfollow. Facebook, Twitter, Flickr integration. (partial) Twitter API support. Fully Restful design, user interface consumes Rest API.	Yes

Note: *Some of the distributed architectures, features and privacy support are adapted from Wikimedia Foundation Inc. (2015).

Table 1 Architectures' summaries with privacy features (continued)

Project name		*Features*	*Privacy support*
		*Architecture type: distributed architecture**	
30	Project Danube[32]	1 Sharing personal data with companies/organisations) 2 Sharing personal data with 'friends' 3 Use of personal data for 'personal applications'	Not found
31	Project Nori[33]	Personal data store (PDS) (a.k.a. personal data locker)	Not found but could be possible centralise control for the user through 'PDS'
32	Psyced[34]	Profiles, chat, micro blogging	Yes by running user's own server so data still resides with users.
33	Pump.io[35]	Stream server supporting social networking capabilities	Yes
34	RedMatrix[36]	Decentralised identity platform, also provides blogs, rich social networking, cloud storage and internet-scale access control/privacy	Extensive
35	Retroshare[37]	Private messaging	
36	Safebook (Cutillo et al., 2010)		Extensive, including communication untraceability
37	Salut Ã Toi[38]	Multi-frontends, micro blogging, group micro blogging, file sharing, games, XMPP client	through XMPP groups, presence authorisation
38	SMOB[39]	Micro blogging	Not found
39	Social-Igniter[40]	Friends, places, status, comments, modular apps (messages, blog, cart, media), themes, mobile themes, 3rd party integration (Facebook, Twiter, YouTube), editable widgets,	Yes
40	SocialRiver[41]		Private messaging, privacy controls
41	SocialZE[42]		Yes
42	SocknetProvider-FoolishMortal.org[43]	profiles, messaging, enables internet content sharing	No
43	Sone[44]	Micro blogging, media library, decentralised spam protection	Yes, multiple anonymous identities, private messages via the Freemail plugin with Forward_secrecy
44	Sparkleshare[45]	Collaboration and sharing tool inspired by dropbox	Yes, encryption option

Note: *Some of the distributed architectures, features and privacy support are adapted from Wikimedia Foundation Inc. (2015).

Table 1 Architectures' summaries with privacy features (continued)

Project name		*Features*	*Privacy support*
		*Architecture type: distributed architecture**	
45	Tent[46]	Profiles, developer-extensible post and profile types, data import, groups, privacy controls, content versioning	Yes granular permissions (access control lists for all content)
46	Thimbl[47]	Micro blogging	Not found
47	Twister[48]	Micro blogging	Yes end-to-end encryption for private messages
48	Weestit[49]		Yes
		Architecture type: third party architecture	
1	Smoke Screen (Cox et al., 2007)	Use trusted third party or broker	Yes through trusted third party or broker
		Architecture type: Peer-to-peer or ubiquitous architecture	
1	PeerSoN (PeerSoN, 2016)	Establish social networking by identifying another user who is both close by and using another mobile device	Yes peer-to-peer trust
2	Opera Unite (Opera Software ASA, 2011)		
		Architecture type: wireless sensor network architecture	
1	BlueTrust (Markides and Coetzee, 2008)	Semantic Web and mobile ad-hoc networks, a simple number as the trust mechanism	Ensure privacy based on a simple number as the trust mechanism
2	Serendipity (Eagle and Pentland, 2005)	Friendship initiation system	Bluetooth-based trust systems for friends
3	WhoZThat (Beach et al., 2008)	Users can also initiate friendship by sharing their social networking ID	Not yet

Note: *Some of the distributed architectures, features and privacy support are adapted from Wikimedia Foundation Inc. (2015).

3 Architecture models

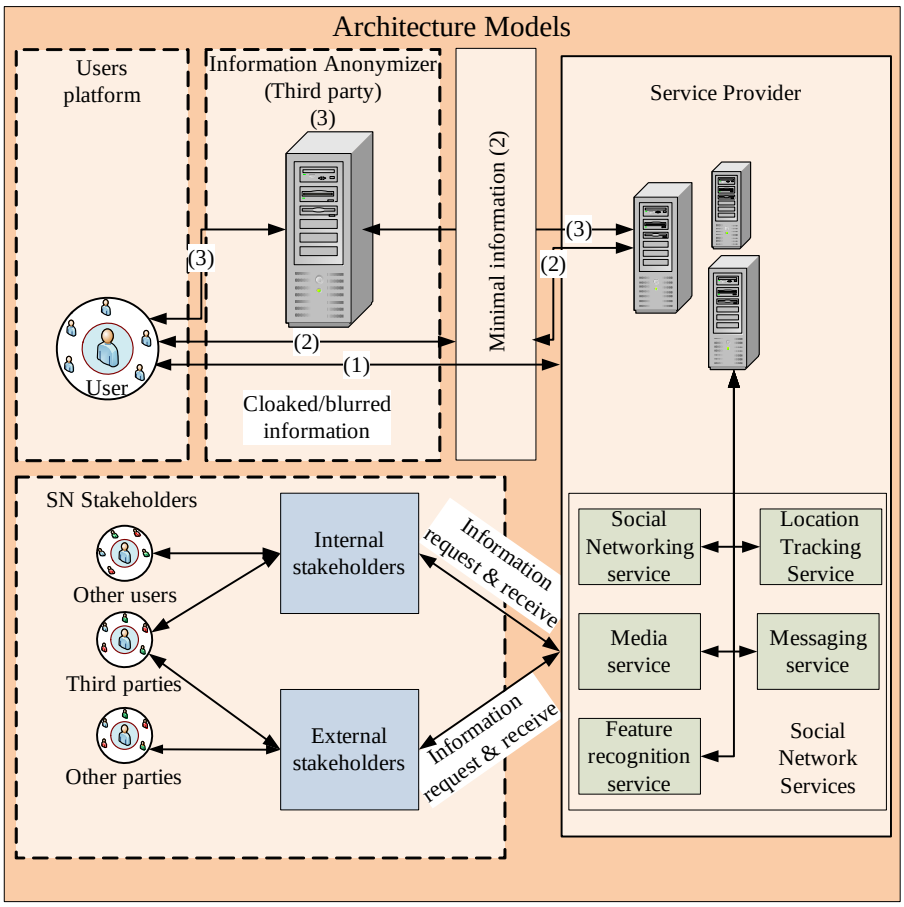
The purpose of this section is to define the business architecture models and various privacy models, in order to scope the SN to which the privacy-preserving architecture is applicable. The various privacy models available to protect user information between connected stakeholders are the access control model, minimal information sharing, and the third party model.

3.1 Access control model

The access control model is used to ensure privacy by setting up access control for the SN stakeholders. Figure 1 shows this model where users can control access for service providers and stakeholders through a service provider. For the 'access control model', the target architecture uses the privacy model to control access for the data types [Refer

Figure 1, ignoring the minimal information (2) and the information anonymiser's third party (3) component]. Service providers will pre-define the privacy model and SN users have the flexibility to customise it.

Figure 1 Architecture models (see online version for colours)



However, access control is inherently inadequate to address privacy on the internet. The access control paradigm can be described as discretionary access control, as the 'need-to-know' access model, or as role-based access control (Fong et al., 2009). With this model, stakeholders have little control over how their data is used and accessed (Kagal and Abelson, 2010).

3.2 Minimal information sharing model

The minimal information sharing model performs cryptographic techniques such as join and intersection operations, or secures information. However, the computational cost and

the inability to facilitate other queries make this paradigm unsuitable for real time applications (Chow, 2010).

Other solutions, such as 'de-identification' by removing or modifying PII (that is, social security numbers, driver license numbers or financial accounts) and 're-identification' whenever individual and sensitive information is needed (Narayanan and Shmatikov, 2010), can be part of the system architecture. While differential privacy can ensure good privacy protection in the architecture, it is inadequate with respect to data accessibility.

For the 'Minimal information sharing', the target architecture uses the minimal information sharing component inside the service provider platform [Refer Figure 1, ignoring the direct link between users' platform and service provider information (1) and anonymiser's third party (3) component]. Service providers will store partial information based on their preferred minimal sharing algorithm.

3.3 Third party model

The third party model engages a third party to protect SN user information. This model is further divided into the untrustworthy third party and trustworthy third party models. The untrustworthy third party model engages an untrusted third party, which executes queries by collecting secure information from multiple data sources (Chow, 2010). However, the computational cost and the emphasis on processing and securing information from multiple data sources make this paradigm unsuitable for real time applications. The trusty third party model engages a third party trusted by users and acts as a middle layer between the database server and users, to process information (Chow, 2010). This paradigm is already engaged in various location-based services.

For the 'third party model', the architecture uses anonymiser as third party storage to protect user privacy, and consists of three logical technology components: user platform, information anonymiser, and service provider platform [Refer Figure 1, ignoring the direct link between users' platform and service provider information (1) and the minimal information (2) component]. Information anonymiser works as a middle layer between user and service provider.

A trusted third party could be engaged to protect SN user information; however, there are both strong supporters and strong opponents of both the minimal information sharing model, which uses cryptographic techniques to share minimum information (Agrawal et al., 2003a), and trusted third party models, which use a third party server to protect user privacy (Aggarwal et al., 2004; Jefferies et al., 1996).

The computational cost and inability to facilitate the performance of every cryptographic technique query makes the minimal information sharing paradigm unsuitable for real time applications (Chow, 2010). On the other hand, the trusted third party model is already used by existing location privacy techniques (Mokbel et al., 2006; Xu and Cai, 2007, 2008; Peng et al., 2008; Kalnis et al., 2007; Gruteser and Grunwald, 2003; Gedik and Liu, 2008; Chow et al., 2006; Chow and Mokbel, 2007; Bamba et al., 2008; Beresford and Stajano, 2003). It is commercially engaged in ensuring online user privacy in: PayPal (1999–2016) for buying and selling products; anonymiser[50] for anonymous surfing; and in Wi-Fi protection. However, the third party must be completely and highly trusted for the storage and sharing of information (Agrawal et al., 2003a).

4 Privacy protection mechanisms

SN users have ultimate ownership of their personal data and can take control to ensure its privacy. Users may share private information with friends and (often) strangers. In so doing, however, major concerns can include being identified by malicious adversaries, and having sensitive relationships revealed (Liu et al., 2010). SN users can, however, manage their personal and private information via a well-informed approach, such as access control for other SN users, or advanced mechanisms such as privacy by friends-of-a-friend prediction. Setting up such privacy protection mechanisms, however, requires significant effort and may lead SN users to accept the default setting; this eventually results in a loss of privacy and loss of control over one's personal information.

4.1 Privacy by access control

Privacy can be seen as a companion to access control for linked members on SNs. Currently, users are required to make a substantial effort in, and devote significant time to, setting up access controls for other individuals in order to protect themselves from privacy breaches.

There are several ways in which users can implement access control:

1 they may be technically adept enough to understand the privacy settings of the SN

2 they might be able to employ a mobile access control list (MACL), a privacy control mechanism which considers 'user attitude', 'user communication history' and 'social aspect' (Lugano and Saariluoma, 2007)

3 they might ensure privacy by using a SN Privacy Wizard template (Fang and LeFevre, 2010), which iteratively captures a limited number of user inputs to assign privacy 'labels' to selected friends and infers a user privacy preference

4 they can utilise 'Privacy Butler', an automated service which monitors privacy policies and filters unwanted activities of connected friends (Wishart et al., 2010)

5 they can avail themselves of the advantages of the capabilities of both the Semantic Web and mobile adhoc networks to ensure privacy based on a simple number as the trust mechanism, such as the 'BlueTrust' system (Markides and Coetzee, 2008); however, trust is more complex and, while regularly available in the Bluetooth range, the system may not provide symmetric or asymmetric trust.

4.2 Privacy by friends-of-a-friend prediction

Privacy can be achieved through new concepts such as 'friends-of-friends', which are based on symmetric or asymmetric scenarios. Privacy by friends-of-a friend prediction has drawn the attention of social researchers who hold that everybody on the planet is connected by 'six degree relationships': your friends belong to a 'one degree relationship'; friends of your friends belong to a 'two degree relationship'; and so on (Cai et al., 2009). The user can apply varying levels of control to the publishing and sharing of information or resources according to the friend-of-a-friend relationship theory.

'Serendipity', for example, is one of the initial Bluetooth-based systems which demonstrates the friendship initiation system (Eagle and Pentland, 2005); however, it

does not utilise WiFi, and is not extendable to Multihop mesh networking similar to 'WhoZThat', which ties mobile smart phones to multiple SNs and initiates friendships by sharing social networking ID (Beach et al., 2008) extended to a context-aware component, such as Music Jukebox; however, linking to SN information may cause future privacy breaches.

Friendships can be hidden but can be revealed by using secure hashing identifiers, and friends-of-a-friend can be matched through hashed key to disclose unilateral friendships. This hidden friendship matching technique avoids privacy-depleting consequences (Preibusch and Beresford, 2009). Raban et al. (2009) describe mechanisms for exploring relationship between strangers, and recommends a 'restrictive profile' for mobile devices. Their investigation incorporates two established theoretical approaches – uncertainty reduction theory (URT) and predicted outcome value theory (POV) – to develop an efficient introduction mechanism between users based on synchronous progressive disclosure of personal information.

Link prediction algorithms to anonymise a dynamic SN is one of the privacy preserving techniques (Bhagat et al., 2010). Bhagat et al. propose clustering methods that consequently provides guaranteed anonymity against adversaries with limited background knowledge; however, the personal information which SN users provide could be misused and tampered with. The connection linking mechanism is also a SN focus area. New connections are always essential elements of a SN; however, it is crucial to choose valid connections.

4.3 Web service federated or distributed perspective

Privacy can be assured through new concepts of web services federation or distribution. The federation or distributed approach can be composed and compared as various approaches such as (GNU social, 2014b):

1 commodity web hosting

2 non-free software

3 federation of servers

4 P2P and DHT

5 social desktop applications

6 in-browser profile and certificates

7 distributed node architecture.

4.3.1 Commodity webhosting

In the commodity web hosting (GNU social, 2014b) approach, web services are often deployed on virtual machines or commodity web hosting. Some examples of this type of service are StatusNet (StatusNet Inc., 2010), and Diaspora (Diaspora, 2010). However, encryption, security and privacy are not safe on virtual machines, and the federation on these machines and servers cannot handle as much traffic as applications using optimised protocols rather than HTTP.

4.3.2 Non-free software

Non-free software-based (GNU social, 2014b) services hide user data and only allow its sharing based on attribute-based encryption; examples are iSocial (iSocial ITN, 2014) and Persona (Baden et al., 2009). However, privacy should not intend to simply hide user data as, although hidden, it will remain in the system and can still become the cause of future privacy incidents.

4.3.3 Federation of servers

Server federation (GNU social, 2014b) approaches use existing server infrastructure and traditional internet architecture; some examples are OneSocialWeb (Vodafone Group, 2010) and XMPP (The Internet Engineering Task Force, 2014). However, these approaches construct services using a certain degree of unencrypted trust in the servers.

4.3.4 P2P/DHT

The P2P and DHT (GNU social, 2014b) approach can be considered as one of the best privacy approaches, as it provides for any social interaction to be end-to-end or group encrypted; some examples of this approach are SNs are PeerSoN (PeerSoN, 2016), Safebook (Cutillo et al., 2010), Friend2Friend (Altruists International, 2001) and Opera Unite (Opera Software ASA, 2011). Some fine-tuned technologies improve their services by concealing the identities of those involved in a communication. However, availability of services is still doubtful since the services need a special strategy for message delivery when a source goes 'offline' with a redundancy between servers, DHT, and/or group communication.

4.3.5 Social desktop applications

Social desktop application (GNU social, 2014b) approaches allow end to end encryption for people and groups without engaging a web browser, and provide richer interactions beyond traditional SNs. Examples of this type of application are Nepomuk for KDE (KDE, 2014) and Social Desktop for KDE (KDE, 2009). While these applications provide computer desktop experience integration, they could be the source of possible privacy issues similar to client-server approach.

4.3.6 In-browser profile and certificates

The In-browser profile and certificates (GNU social, 2014b) approach stores secure user profile locally in the browser and authenticates it at any website; external websites are not able to breach it using Friend of a Friend+Secure Socket Layer (FOAF+SSL) protocol. An example in this category is Lorea (2010). In this approach, FOAF+SSL securely includes a link to a profile request by the web browser, and can be hosted on commodity web hosting. However, this approach includes a layer of complexity without solving the privacy issues; this is because creating a forum and micro blogging requires some sort of hosting, and group encryption is not possible in this approach. The user also needs to surf websites to receive profiles and information updates and there is no real-time notification stream; these are other drawbacks of this approach.

4.3.7 Distributed node architecture

Distributed node architecture (GNU social, 2014b) separates an end user's social node into five components – 'core', 'UI', 'core transports', 'datastore modules', and 'UI transports' – and defines a framework for their interaction; Distnode (GNU social, 2014a) is an example of this category. This approach facilitates end-to-end encryption for people and groups, and the use of transport protocols such as HTTP, XMPP, or PSYC for relaying data between nodes. The approach also allows users to access the same account using various client programs such as Web Browser, Dedicated App and MeMenu. This approach is impressive but there is the possibility of over-design.

Interoperability between these federated services, which is inadequately addressed by the services, can be conquered by various privacy languages. There are also many privacy languages available for representing policies in a human-readable and machine-readable format (Kumaraguru et al., 2007), such as: 'Platform for Privacy Preferences (P3P)' (Cranor et al., 2002b); 'A P3P Preference Exchange Language (APPEL)' (Cranor et al., 2002a); 'Customer Profile Exchange (CPExchange)' (Bohrer and Holland, 2000); 'Privacy Rights Markup Language (PRML)' (Zero-Knowledge Systems Inc., 2004); 'XML Access Control Language (XACL)' (Kudo and Hada, 2000); 'Platform for Enterprise Privacy Practices (E-P3P)' (Karjoth et al., 2003); 'Security Assertion Markup Language (SAML)' (SSTC, 2004); 'Rei' (Kagal et al., 2005); 'eXtensible Access Control Markup Language (XACML)' (OASIS Standard, 2005); 'Enterprise Privacy Authorization Language (EPAL)' (Ashley et al., 2003); 'X-Path Based Preference Language (XPref)'(Agrawal et al., 2003b); 'Declarative Privacy Authorization Language (DPAL)' (Barth et al., 2004); 'Geographic Location Privacy (Geopriv)" (Schulzrinne et al., 2007); and ODRL (Iannella, 2002). However, while these privacy languages are useful for interoperability between new or existing SNs, it is necessary to address the privacy issues for these SN services.

Privacy preservation should not be partial but should ensure end-to-end preservation for SNs. Enterprises and service providers are accountable for the complete protection of the privacy of all SN users. Ensuring privacy at the system design level by using Privacy by Design (PbD) or other technologies can be the future privacy protection solution for SN users.

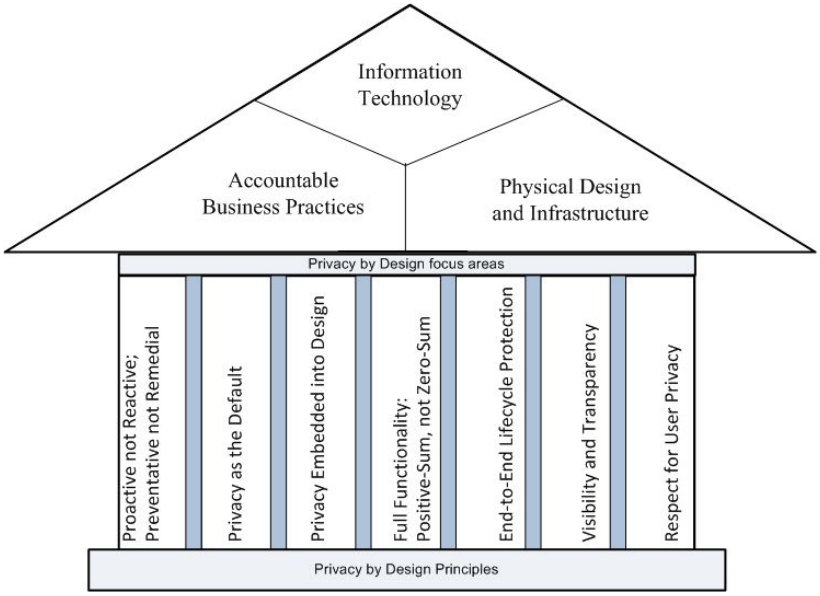
4.4 Privacy by innovative architecture

Researchers have designed various innovative SN architectures to protect privacy in SNs; for example, Matryoshka structure (Cutillo et al., 2010) and Contrail (Studi et al., 2010) provide an innovative and complete solution to preserve privacy, data integrity, data availability and data lookup. P2P architecture is used in the Matryoshka structure, and cloud-based P2P architecture is used in Contrail. However, the feasibility of the P2P SN architecture in terms of availability of data and responsiveness of the system is still an open question; additionally, no existing large-scale SN has been considered for Contrail. One of the specific limitations of the Contrail system is that the pre-trusted assumption between two users implies synchronous mutual trust; however, trust can be asynchronous. Another limitation of Contrail is that it assumes that the cloud is reliable and will not lose any data; however, cloud privacy is another issue for future research.

Other solutions, such as 'de-identification' by removing or modifying PII (such as social security numbers, driver's license numbers or financial accounts) and

'reidentification' whenever individual and sensitive information is needed (Narayanan and Shmatikov, 2010) can be part of the system architecture. Additionally, differential privacy can ensure effective privacy protection in the architecture; however, Narayanan and Shmatikov's (2010) findings might have been much more convincing if the they had considered data accessibility in the proposed process.

Figure 2 PbD principles (see online version for colours)



Source: Cavoukian (2009)

4.5 Privacy protecting principles

There are various principles available for protecting privacy. These principles are Fair Information Principles (FIPs) (United States Federal Trade Commission, 1973), Organization of Economic Cooperation and Development (OECD) Principles (Organisation for Economic Co-operation and Development, 2002), CSA Model Code Principles (CSA Group, 2014), Information Privacy Principles (IPPs) (Office of the Australian Information Commissioner, 1988b), National Privacy principles (NPPs) (Office of the Australian Information Commissioner, 1988a), Nine Architectural Principles (Diamond et al., 2008), and PbD principles (Cavoukian, 2009).

FIPs are included in the national level but do not includes in the federal level, and no safeguards exist to ensure that these principles are implemented (Diamond et al., 2008). Another major weakness is that the principles allow agencies to use private sector data without any appropriate protections from law. OECD Principles are confined to the European Union directive to protect personal data; however, those principles are strengthened, particularly in 'consent' and 'accountability', in the CSA Model Code Principles. The Nine Architectural Principles are designed especially to protect privacy in

a networked environment (Diamond et al., 2008). IPPs manage personal information for government agencies, whereas the NPPs regulate the private sector.

4.5.1 Privacy by design

The term PbD (Cavoukian, 2009) was conceived by Dr. Ann Cavoukian in early 1990. Gradually, she has distilled PbD into seven key principles; however, these principles remain at the conceptual stage. To comply with the PbD concept and to ensure privacy, a system needs to be systematic, predictable and repeatable (Cavoukian, 2009). Figure 2 shows the PbD principles. Refer to Table 2 for the PbD principles details and examples.

Table 2 PbD principles and examples

PbD principle	*Analysis and examples*
1 Proactive not reactive; preventative not remedial	Privacy protection comes before-the-fact, not after. This principle dictates that information privacy will be considered and ensured before problems arise. Conducting a privacy impact assessment (PIA) is one of the early approaches to preventing privacy threats (Cavoukian and Spencer, 2010); however, the PIA should be repeated and updated after a period of time (say, half-yearly or yearly). Cavoukian and Spencer (2010) also demonstrate a practical case study, utilising PbD principles pro-actively. In this case study, the Ontario Health Study team pro-actively provided de-identified protected information to assist researchers to enable further comprehensive studies of cancer, vascular diseases, and other chronic diseases. The case study also found that physical privacy another proactive consideration as important as information privacy; if the former is not provided, the respondents may not feel comfortable in providing the protected information. Radio frequency identification (RFID) within the health sector is an area that requires privacy measures (Cavoukian, 2009). Cavoukian (2009) suggested that RFID usage in this sector should not be linked to personal identifiers so as to avoid potential short or long-term threats to personal privacy proactively. Williams and Weber-Jahnke (2010) provide three solutions to privacy breaches in healthcare SNs. These are: the use of automated queries to detect false user accounts; the development of improved business processes to detect credentialed users; and preventing users from locating hidden network information. These solutions can also be considered in other privacy-invasive areas. As a precaution, users should be well informed about the privacy policy and conditions and terms of use of a SN. They should always be told: 'who' will access their private information; 'how' it will be accessed; and 'where' it will be accessed in the SN. However, a lengthy, text-based privacy policy is not sufficient as the user seldom reads such privacy policies (Kelley et al., 2009); rather, service providers should provide graphical-based understandable privacy policies. Preventive rather than remedial privacy features must be an aim for SNs; however, raising user awareness of privacy issues, as well as the provision of proactive features, can assist in improving privacy before breaches occur. The user must be aware that their SN profile is linked to their real world social identity; they should also be aware that failing to protect their own privacy will eventually also increase the threat and risk to the privacy of friends and family.

Table 2 PbD principles and examples (continued)

PbD principle	*Analysis and examples*
2 Privacy as the default	Privacy as the default ensures that no action is required on the part of the individual to protect their privacy; it is built into the system by default, and information use and collection is determined by a respect for individual privacy. Privacy should be built into the SN system to protect a user's private information and to ensure privacy by default. This is necessary as personal information in the user's SN profile can represent and/or replicate their social identity. Thus, if the service provider reveals the user profile information to a third party, it may be harmful to the user. There are many ways in which service providers can build automatic privacy features into the systems; for examples: 1 Privacy settings should not include an opt-in approach to automatically disclose private and protected information (The removal of this opt-in approach might avoid unwanted disclosure problems) (Cavoukian, 2009). 2 When the user initially engages in a SN, the default settings must not disclose any protected information such as location information, e-mail address, date of birth or financial information (A possible approach could be to use a pseudonym for every user; hence, the service providers would not be able to access the actual name of that person and misuse their protected information). 3 Friendship requests from other users must not be accepted or approved by default in the SN. 4 Location-based features should be deactivated by default in the mobile SNs (Extra care should be taken while activating location-based services and information in a mobile SN; users may not be aware of the activated location-based service, which can eventually lead to a privacy breach). 5 Providers can create a user-centric identity management infrastructure (If there are any changes in the system, the user has to approve the updates and receive feedback for the changes in the system. After some time, the user might feel comfortable and satisfied with the system's privacy; at that point, they might choose to change their privacy settings (Ahern et al., 2007) and relax their privacy requirements). 6 Information can be generalising after a period of time (Williams and Weber-Jahnke, 2010) by automatically generalising the accessible information to an inactive connection (for example, the connection between a user and health care provider can be degraded over many years in a healthcare SN). Built-in default privacy might be advantageous; however, it could cause the user to be relaxed about their privacy and to fail to verify the system's default privacy features. While it is unlikely, in reality, these settings themselves can be a source of various privacy breaches (such as leaking information to an untrustworthy third party). Therefore, the user should be well informed about the status of the default privacy settings.

Table 2 PbD principles and examples (continued)

PbD principle	*Analysis and examples*
3 Privacy embedded into design	Privacy embedded in design is a key issue in implementing privacy. One of the privacy-invasive systems – biometric encryption – could utilise PbD principles to provide privacy and ensure full functionality (Cavoukian and Stoianov, 2007). Such systems should be designed so that they store only the biometrically encrypted code, rather than the biometric information itself. In this approach, third parties will have less interest in collecting and accumulating actual biometric information; thus, embedded privacy in the design will protect user information and the required functionality will be ensured. Another privacy-invasive area is Video Surveillance which seeks to ensure public safety with respect to governance, but at the expense of the privacy of law-abiding citizens (Cavoukian, 2009). To address this privacy issue, one approach could be to publish general information on a website to inform citizens about the locations of public video surveillance and reasons for its installation. Another approach could be to strictly control the PII, such as face images and location data, and thus avoid unauthorised access. Williams and Weber-Jahnke (2010) suggest two privacy mechanisms that can be incorporated at the system design level. The first mechanism involves the adoption of only those architectures that allow for anonymity; the other requires that third parties comply with the system's user privacy policy. In these ways, service providers can ensure the availability of information required by interested third parties through provided interfaces rather than direct access, while at the same time ensuring user privacy.
4 Full functionality: positive-sum, not zero-sum	It is possible to have both such as privacy vs. security. The principle underpinning the methodology is how to create full functionality while protecting individual privacy. The future of privacy-preserving SN applications is expected to be a win-win scenario: service providers' business models will not be destroyed (Weiss, 2009), and user privacy will be protected. Weiss (2009) proposes a privacy threat model that can be used to enhance information privacy to protect PII. In the proposed model, the SN application user will have fine-grained three dimensional controls over their PII. Such control is one of the fundamental requirements of major privacy laws in Europe. Obviously, a positive-sum paradigm is achievable in the system design (Cavoukian, 2009). There is a myth that one goal is achieved at the expense of another. This is not necessarily true, especially in the health sector; Cavoukian (2009), for example, demonstrate a framework which provides de-identified e-health records with a low level of re-identification risk. Hence, privacy and data quality is ensured. Similarly, biometric encryption ensures full functionality (Cavoukian, 2009) by storing biometrically encrypted codes only, rather than storing biometric information; hence, third parties will not be interested in collecting and storing actual biometric images. In addition, Cavoukian (2009) also examines (along with Bering Media's Technology), the application of the positive-sum paradigm so that internet service providers (ISPs) can ensure they provide full functionality with zero disclosure of subscribers' PII (Cavoukian and Emam, 2010). With this innovative double-blind privacy architecture, ISPs never learn the physical location of the IP address for precise geo-locations (such as postal code or ZIP+4 information), and advertisers are referred to an identifier number without any actual details.

Table 2 PbD principles and examples (continued)

PbD principle	*Analysis and examples*
5 End-to-end lifecycle protection	PbD ensures cradle-to-grave, lifecycle management of information. The principle underpinning the assessment is how to secure information along with privacy. Williams and Weber-Jahnke (2010) describe possible solutions to end-to-end protection in the healthcare SN. One solution is anonymising network information if requested by interested parties such as governments, researchers or advertising companies. Another possible solution can be a fine-grained access control mechanism where user information can be accessed by third parties; in this way, others can access general information rather than actual information. Another example of an end-to-end protection solution is given by Narayanan and Shmatikov (2010). In this approach, individual and sensitive information can be 'de-identified' by removing or modifying PII (such as social security numbers, drivers' license numbers or financial accounts). Narayanan and Shmatikov (2010) claim that differential privacy, de-identification and re-identification, are effective privacy protection. However, their findings might have been more convincing if they had considered data accessibility in the proposed process. Custodian or service providers must have procedures to securely dispose of personal records such as health records, SN member information or other PII in a timely manner. The user information can be discontinued for particular reasons (for example, someone is deceased) (Facebook Inc., 2015). Furthermore, precautionary measures can be taken to simplify the end-to-end protection process. One such precautionary measure could be the use of the physical security features already available for mobile phones; for example, not storing the password, never leaving the phone unattended, and reporting immediately if it is stolen. One precautionary measure to deal with the possibility of the latter can be to encrypt user information when transferring it to hand-held devices. Information can also be encrypted when the information is being transferred to a centralised server, and decrypted when it is transferred to a hand-held device to avoid communication privacy breaches.
6 Visibility and transparent	Trust but verify. The principle underpinning the investigation is how the accountable organisation will be open and honest with individual privacy. The accessing of information must be visible and transparent. For example, RFID cy technology can enhance visibility and transparency; however, wherever possible, it is necessary to minimise the identifiable, observable and linkable RFID information to prevent future threats to privacy (Cavoukian, 2009). Additionally, the individual participant should be informed of any changes, so as to make the RFID system as open and transparent as possible. The user should have a transparent view of how any system is collecting their information. Delgado et al. (2010) propose a solution where a SN application indicates how it will collect the user's information. In this way, the user will know which applications can access which information from their profile. Williams and Weber-Jahnke (2010) claim that mobile SN service providers might provide an unclear view of their privacy policies since these contain only text. They also claim that visibility has not been a strong point of SNs, and suggest that diagrams or interactive tools could be incorporated to increase the visibility or transparency of their policies. Nevertheless, a user could still have an unclear view of unexpected information propagation across the SN. It is also unlikely that the user will dispute the long textual privacy policies as this would mean that they would not be able to use the system. In most cases, a typical user does not read such policies (Kelley et al., 2009).

Table 2 PbD principles and examples (continued)

PbD principle	*Analysis and examples*
7 Respect for user privacy	Keep the system user-centric. The principle underpinning the investigation is how to share, disclose or access, rectify, delete, and block information that is consistent with respect to individual privacy.
	Roig (2010) concludes that privacy enhanced technology (PET) and transparency-enhancing technology (TET) are needed to be incorporated in the initial design level to ensure user privacy. Roig (2010) also claims that PETs should not be limited to anonymity, pseudonymity, unlinkability or unobservability, but should also be required to include transparency, automatic compliance assurance functions, and proactive techniques for risk analysis. Furthermore, TETs such as 'sticky policies' should provide clear information mechanisms and cross-disciplinary professions such as lawyers and designers are also required to work together to address the privacy issues in SN applications.
	SN service providers should take the necessary steps to decrease the user burden and to respect user privacy (Williams and Weber-Jahnke, 2010). They should provide an interactive user interface for controlling privacy settings; in this way, the user can edit, hide, or delete their personal information. If anyone sees their profile or personal information without their consent, the user will know, and be cautious when publishing information in future. In SNs, user-centric identity management can be one means of enabling users to protect their own privacy details.

5 Enterprise architecture methodologies

There are a number of enterprise architecture methodologies available to implement privacy-preserving architectures. However, as many as 90% of these have focussed on four methodologies:

1 the Zachman framework for enterprise architectures

2 the open group architectural framework (TOGAF)

3 the federal enterprise architecture (FEA)

4 the Gartner methodology (Sessions, 2007).

5.1 The Zachman framework

The Zachman framework is self-described as a 'framework'; however, Sessions (2007) defined this framework as 'ataxonomy' since it organises architectural artefacts such as design documents, specifications, and models as taxonomy. The Zachman (2008) framework is typically depicted as six functional foci (data, function, network, people, time, and motivation) from the perspective of six major players of an organisation (planners, owners, designers, builders, subcontractors, and enterprises); these are represented in a 6 × 6 'matrix', with the Communication Interrogatives as Columns and the Reification Transformations as Rows (Sessions, 2007).

5.2 The open group architectural framework

TOGAF (The Open Group, 2013) is also known as a 'framework'; however, Sessions (2007) defines the framework as 'a process' since the architecture development method (ADM) is used as a process for creating an enterprise architecture. TOGAF has a detailed method and set of supporting resources for developing an enterprise architecture.

5.3 The federal enterprise architecture

The FEA can be viewed as either an implemented enterprise architecture, or as a proscriptive methodology for creating an enterprise architecture (Sessions, 2007). FEA (United States Executive Office, 2012) provides principles and standards within and between agencies and external stakeholders across the Federal Government to develop business, information, and technology architectures.

5.4 The Gartner methodology

A well-known and leading organisation, Gartner Inc. (2014) has developed the Gartner methodology and many well-qualified specialist communities which encourage collaboration and best practice in technology research. Sessions (2007) describes the Gartner methodology as 'an enterprise architectural practice', as explored in Gartner Enterprise Architecture Process: Evolution 2005 (Bittler and Kreizmann, 2005).

5.5 Comparison of enterprise architecture methodologies

Sessions (2007) suggests a much more systematic approach to identify which architecture methodology is appropriate for an enterprise, and to distinctly differentiate between various enterprise architecture methodologies so as to establish criteria and ratings for each methodology. Sessions evaluated the methodologies based on 12 criteria and a ranking of 1–4 for each criterion. If a criterion does 'a very poor job' in that area, then the methodology scores '1'; it scores '2' for 'an inadequate job', '3' for 'an acceptable job', and '4' for 'a very good job'. Table 3 (adapted from Sessions, 2007) shows these criteria and ratings for enterprise architecture methodologies. Sessions recommends working through the criteria and determining the appropriate architecture methodology, as each has its strengths and weaknesses, and none is complete; for example, TOGAF has scored '4 – a very good job' for 'process completeness'.

Table 3 Criteria and ratings for enterprise architecture methodologies

Criteria	*Ratings*			
	Zachman	*TOGAF*	*FEA*	*Gartner*
Taxonomy completeness	4	2	2	1
Process completeness	1	4	2	3
Reference-model guidance	1	3	4	1
Practice guidance	1	2	2	4
Maturity model	1	1	3	2

Source: Adapted from Sessions (2007)

Table 3 Criteria and ratings for enterprise architecture methodologies (continued)

Criteria	*Ratings*			
	Zachman	*TOGAF*	*FEA*	*Gartner*
Business focus	1	2	1	4
Governance guidance	1	2	3	3
Partitioning guidance	1	2	4	3
Prescriptive catalogue	1	2	4	2
Vendor neutrality	2	4	3	1
Information availability	2	4	2	1
Time to value	1	3	1	4

Source: Adapted from Sessions (2007)

6 Conclusions

After analysing the literature of privacy issues in SNs architectures, we can observe that this area is still an open domain for research and analysts, researchers and SN users are expressing their concerns about these issues, and are engaging in attempts to mitigate these concerns. Analysing the literature, we have discovered points that remain open problems in research. To summarise, there is a need for a more fine-grained privacy protecting architecture that, in addition to being developed using appropriate privacy protecting principles and research methods, should be

1 *Distributed/federated web service approach:* The contemporary distributed/web service federated approach for incorporating PbD to protect user privacy is recommended for SN services. This approach has established specific mechanisms for individuals, business, developers, government and academia (Federated Social Web Community Group, 2005). Individual users can decide where to store their data, which tools and features they use for their services, which provider they prefer, and can specify the technology which is used to store their data in their own individual storage. In this way, they can accomplish jurisdiction and rights over their own data (Federated Social Web Community Group, 2005). Although extensive research has been carried out on the web service federated/distributed approach, there are still challenges that are not adequately covered by this approach. Section 4.3 explains and compares some of these challenges. The key limitation of the web service federated/distributed approach is that providers may need to develop completely new services to achieve interoperability from the outset.

2 *Engaged preferred model:* In the target architecture, the service provider should facilitate the SN services and functions based on preferred architecture model. In the access control and minimal information sharing model, service providers should store user information based on PbD principles. In the third party model, the service provider works as a third party storage device. The service provider contains information, but is unable to access user information directly. The service provider will store cloaked information provided in the 'information anonymiser', and will facilitate user information based on the user's privacy positions.

3 *Adapted PbD principles:* To alleviate user burden and to ensure their full ownership of private information, the latest design paradigm, PbD, can be adapted into SNs. Embedding privacy directly into the 'design and operation' (Cavoukian and Prosch, 2011) of a system can ensure the protection of privacy from the outset (Cavoukian and Prosch, 2011). In fact, incorporating privacy at the design stage can ensure SN users' autonomy by default. This will reduce the burden on users and encourage an increase in social networking. This research suggests that seven PbD principles should be incorporated into a system at the design stage. Indeed, one of the specific objectives of this research is to encourage the engagement of privacy requirements in the system design stage for two reasons:

   a system development costs increase substantially in later stages of system development, so it is useful if privacy can be incorporated at this initial stage

   b privacy functionality can easily be engaged in the initial design stage, while it is extremely difficult to incorporate privacy in the later stages of system development.

4 *Adapted TOGAF:* Since any architecture design is tremendously complex, this research suggests the TOGAF, based on a comparison of the top four enterprise architecture methodologies (Refer to Section 5 which includes a comparison of enterprise architecture methodologies, and lists a set of criteria for selecting appropriate architecture methodologies). All the studies reviewed so far, however, suffer from the fact that none is complete. Prior studies have noted the importance of selection of the TOGAF as the architecture methodology for SN services. This research recommends the TOGAF to develop the SN privacy framework for three reasons. Firstly, TOGAF (The Open Group, 2013) scores the three criteria of process completeness, vendor neutrality, and information availability as necessary to promote sharing and disclosure for SN users (Spiekermann and Cranor, 2009; Tan et al., 2012; Wallbridge, 2009). Secondly, the TOGAF enables frameworks to be tailor-made. Lastly, TOGAF offers 'boundaryless information flow' and open systems implementation (The Open Group, 2013).

None of the prior studies found met all of these requirements, as their focus had not largely been on establishing the architecture to ensure that privacy is a feature of SNs. It needs to: be a fundamental element that is considered and embedded in the initial design stages of the SN development process; exist by default within a system; and be considered throughout the system's life-cycle. Controlling access to information from the design stage will eliminate the need for retrospectively dealing with privacy breaches after they have already caused significant personal embarrassment and/or damage. Since privacy is a basic issue for SNs as users will terminate their usage if they feel unsafe in the SN environment.

Acknowledgements

NICTA is funded by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Centre of Excellence Program.

References

Aggarwal, C.C. (2011) 'An introduction to social network data analytics', English, in Charu C. Aggarwal (Ed.): *Social Network Data Analytics*, pp.1–15, Springer, USA, ISBN: 978-1-4419-8461-6. DOI: 10.1007/ 978-1-4419-8462-3_1. URL: http://dx.doi.org/10.1007/978-1-4419-8462-3_1 (accessed 25 January 2016).

Aggarwal, G., Bawa, M., Ganesan, P., Garcia-Molina, H., Kenthapadi, K., Mishra, N., Motwani, R., Srivastava, U., Thomas, D. and Widom, J. (2004) 'Vision paper: enabling privacy for the paranoids', in *Proceedings of the Thirtieth International Conference on Very Large Data Bases*, Vol. 30, VLDB Endowment, pp.708–719.

Agrawal, R., Evfimievski, A. and Srikant, R. (2003a) 'Information sharing across private databases', in *Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data*, ACM, pp.86–97.

Agrawal, R., Kiernan, J., Srikant, R. and Xu, Y. (2003b) 'An XPath-based preference language for P3P', in *Proceedings of the 12th International Conference on World Wide Web*, ACM, pp.629–639.

Ahern, S., Eckles, D., Good, N.S., King, S., Naaman, M. and Nair, R. (2007) 'Over-exposed?: Privacy patterns and considerations in online and mobile photo sharing', in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, pp.357–366.

Altruists International (2001) *Friend2Friend* [online] http://www.altruists.org/projects/ge/ff/ (accessed 4 January 2016).

Ashley, P., Hada, S., Karjoth, G., Powers, C. and Schunter, M. (2003) 'Enterprise privacy authorization language (EPAL 1.2)', in *Submission to W3C*, p.1.

Baden, R., Bender, A., Spring, N., Bhattacharjee, B. and Starin, D. (2009) 'Persona: an online social network with user-defined privacy', *ACM SIGCOMM Computer Communication Review*, Vol. 39, No. 4, pp.135–146, ACM.

Bamba, B., Liu, L., Pesti, P. and Wang, T. (2008) 'Supporting anonymous location queries in mobile environments with privacygrid', in *Proceedings of the 17th International Conference on World Wide Web*, pp.237–246, ACM.

Barth, A., Mitchell, J.C. and Rosenstein, J. (2004) 'Conflict and combination in privacy policy languages', in *Proceedings of the 2004 ACM Workshop on Privacy in the Electronic Society*, ACM, pp.45–46.

Beach, A., Gartrell, M. and Han, R. (2009) 'Solutions to security and privacy issues in mobile social networking', in *Computational Science and Engineering, 2009. CSE'09. International Conference on*, Vol. 4, IEEE, pp.1036–1042.

Beach, A., Gartrell, M., Akkala, S., Elston, J., Kelley, J., Nishimoto, K., Ray, B., Razgulin, S., Sundaresan, K., Surendar, B., Terada, M. and Han, R. (2008) 'Whozthat? Evolving an ecosystem for context-aware mobile social networks', *IEEE Network*, Vol. 22, No. 4, pp.50–55.

Beresford, A.R. and Stajano, F. (2003) 'Location privacy in pervasive computing', *IEEE Pervasive Computing*, Vol. 1, pp.46–55.

Bhagat, S., Cormode, G., Krishnamurthy, B. and Srivastava, D. (2010) 'Prediction promotes privacy in dynamic social networks', in *Proceedings of the 3rd Conference on Online Social Networks*, pp.6–6.

Bittler, R.S. and Kreizman, G. (2005) 'Gartner enterprise architecture process: evolution 2005', in *G00130849*, Gartner, Stamford, CT, pp.1–12.

Blarkom, G.W.Van, Borking, J.J. and Olk, J.G.E. (2003) 'Handbook of privacy and privacy- enhancing technologies', in *Privacy Incorporated Software Agent (PISA) Consortium*, The Hague.

Bohrer, K. and Holland, B. (2000) *Customer Profile Exchange (cpexchange) Specification* [online] http://xml.coverpages.org/ cpexchangev1_0F.pdf (accessed 4 January 2016).

Bortoli, S., Bouquet, P. and Palpanas, T. (2009) 'Social networking: power to the people', in Papers presented in *W3C Workshop on the Future of Social Networking Position*, January, Barcelona.

Breslin, J., Decker, S., Hauswirth, M., Hynes, G., Le Phuoc, D., Passant, A., Polleres, A., Rabsch, C. and Reynolds, V. (2009) 'Integrating social networks and sensor networks', in *W3C Workshop on the Future of Social Networking* [online] http://www.w3.org/2008/09/msnws/papers/sensors.html (accessed 25 January 2016).

Cáceres, R., Cox, L., Lim, H., Shakimov, A. and Varshavsky, A. (2009) 'Virtual individual servers as privacy-preserving proxies for mobile devices', in *Proceedings of the 1st ACM Workshop on Networking, Systems, and Applications for Mobile Handhelds*, ACM, pp.37–42.

Cai, Y-L., Wang, W-D., Gong, X-Y., Chen, C-F. and Ma, J. (2009) 'Sharing information with controllable precision by distance measuring in mobile social network', in *Wireless Communications, Networking and Mobile Computing, 2009. WiCom'09. 5th International Conference on*, IEEE, pp.1–4.

Camenisch, J., Groß, T. and Heydt-Benjamin, T.S. (2009) 'Accountable privacy supporting services', *Identity in the Information Society*, Vol. 2, No. 3, pp.241–267.

Cavoukian, A. (2009) *PRIVACY BY DESIGN. . . TAKE THE CHALLENGE* [online] https://www.ipc.on.ca/images/Resources/PrivacybyDesignBook.pdf (accessed 25 January 2016).

Cavoukian, A. and Emam, K.E. (2010) *A Positive-Sum Paradigm in Action in the Health Sector* [online] https://www.ipc.on.ca/images/Resources/positive-sum-khalid.pdf (accessed 27 January 2016).

Cavoukian, A. and Prosch, M. (2011) *Privacy by ReDesign: Building a Better Legacy* [online] http://www.itbusiness.ca/blog/ privacy-by-redesign-building-a-better-legacy/20400 (accessed 6 January 2016).

Cavoukian, A. and Spencer, P.C. (2010) *Ontario Health Study Assessment Centres A Case Study for Privacy by Design* [online] http://www.privacybydesign.ca/content/uploads/2010/07/ont-health-assess.pdf.

Cavoukian, A. and Stoianov, A. (2007) *Biometric Encryption: A Positive Sum Technology that Achieves Strong Authentication, Security AND Privacy*. Tech. rep. [online] http://www.ipc.on.ca/images/Resources/up-bio_encryp_ execsum.pdf (accessed 27 January 2016).

Chatterjee, S. (2013) *Ad Hoc Networking Based on Content and Location*. US Patent 8,386,620.

Chow, C-Y. and Mokbel, M.F. (2009) 'Privacy in location-based services: a system architecture perspective', *Sigspatial Special*, Vol. 1, No. 2, pp.23–27.

Chow, C.Y. (2010) *Privacy-Preserving Location-Based Services*, PhD thesis, University of Minnesota.

Chow, C-Y. and Mokbel, M.F. (2007) 'Enabling private continuous queries for revealed user locations', in *Advances in Spatial and Temporal Databases*, Springer, pp.258–275.

Chow, C-Y., Mokbel, M.F. and Liu, X. (2006) 'A peer-to-peer spatial cloaking algorithm for anonymous location-based service', in *Proceedings of the 14th Annual ACM International Symposium on Advances in Geographic Information Systems*, ACM, pp.171–178.

Cox, L.P., Dalton, A. and Marupadi, V. (2007) 'SmokeScreen: flexible privacy controls for presence-sharing', in *Proceedings of the 5th International Conference on Mobile Systems, Applications and Services (MobiSys '07)*, ACM, p.245.

Cranor, L., Langheinrich, M. and Marchiori, M. (2002a) *A P3P Preference Exchange Language 1.0 (APPEL1. 0)*, in W3C Working Draft 15.

Cranor, L., Langheinrich, M., Marchiori, M., Presler-Marshall, M. and Reagle, J. (2002b) *The Platform for Privacy Preferences 1.0 (P3P1. 0) Specification*, in W3C recommendation 16.

CSA Group (2014) *CSA Model Code Principles* [online] http://www.csagroup.org/global/en/legal/privacy/csa-group-privacy-statement (accessed 27 January 2016).

Cutillo, L.A., Molva, R. and Strufe, T. (2010) 'Privacy and identity management for life', in Michele Bezzi, Penny Duquenoy, Simone Fischer-Hübner, Marit Hansen and Ge Zhang (Eds.): *5th IFIP WG 9.2, 9.6/11.4, 11.6, 11.7/PrimeLife International Summer School, Nice, France, September 7–11, 2009, Revised Selected Papers*, Chap. on the Security and Feasibility of Safebook: A Distributed Privacy-Preserving Online Social Network, pp.86–101, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN: 978-3-642-14282-6, DOI: 10.1007/978-3-642-14282-6_7 [online] http://dx.doi.org/10.1007/978-3-642-14282-6_7.

Delgado, J., Rodriguez, E. and Llorente, S. (2010) 'User's privacy in applications provided through social networks', in *Proceedings of Second ACM SIGMM Workshop on Social Media*, ACM, Firenze, Italy, pp.39–44, DOI: 10.1145/1878151. 1878163.

Diamond, C., Goldstein, M., Lansky, D. and Verhulst, S. (2008) 'An architecture for privacy in a networked health information environment', *Cambridge Quarterly of Healthcare Ethics*, Vol. 17, No. 4, pp.429–440, ISSN: 1469-2147, DOI: 10 . 1017 / S0963180108080559 [online] http://journals.cambridge.org/article_S0963180108080559 (accessed 27 January 2016).

Diaspora (2010) *Diaspora* Alpha* [online] https://joindiaspora.com/ (accessed 6 January 2016).

Eagle, N. and Pentland, A. (2005) 'Social serendipity: mobilizing social software', in *Pervasive Computing, IEEE*, Vol. 4, No. 2, pp.28–34, ISSN: 1536-1268. DOI: 10.1109/MPRV. 2005.37.

Ekler, P. and Lukovszki, T. (2010) 'Experiences with phonebook-centric social networks', in *Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE*, IEEE, pp.1–5.

Facebook Inc. (2015) *Facebook* [online] http://www.facebook.com (accessed 5 January 2016).

Fang, L. and LeFevre, K. (2010) 'Privacy wizards for social networking sites', in *Proceedings of the 19th International Conference on World Wide Web (WWW '10)*, ACM, pp.351–360.

Federated Social Web Community Group (2005) *Federated Social Web* [online] http://www.w3.org/community/fedsocweb/ (accessed 28 January 2016).

Fong, P.W.L., Anwar, M. and Zhao, Z. (2009) 'A privacy preservation model for Facebook-style social network systems', in *Computer Security – ESORICS 2009*, Springer, pp.303–320.

Gartner Inc. (2014) *Gartner Methodology* [online] http://www.gartner.com/technology/about.jsp (accessed 7 January 2016).

Gedik, B. and Liu, L. (2008) 'Protecting location privacy with personalized k-anonymity: architecture and algorithms', in *Mobile Computing, IEEE Transactions on*, Vol. 7, No. 1, pp.1–18, ISSN: 1536-1233, DOI: 10.1109/TMC.2007.1062.

GNU social (2014a) *Distnode* [online] https://web.archive.org/web/20140320155703/ https://gitorious.org/ social/pages/Distnode (accessed 7 January 2016).

GNU social (2014b) *Project Comparison* [online] https://web.archive.org/web/20150510004148/ https://gitorious.org/social/pages/ProjectComparison (accessed 7 January 2016).

Gruteser, M. and Grunwald, D. (2003) 'Anonymous usage of location-based services through spatial and temporal cloaking', in *Proceedings of the 1st International Conference on Mobile Systems, Applications and Services*, ACM, pp.31–42.

Hodge, M.J. (2006) 'The Fourth amendment and privacy issues on the new internet: Facebook.com and myspace.com', *Southern Illinois University Law Journal*, Fall, Vol. 31, p.95.

Iannella, R. (2002) *Open Digital Rights Language (ODRL) Version 1.1*, in W3c Note.

iSocial ITN (2014) *iSocial: Decentralized Online Social Networks* [online] http://isocial-itn.eu/ (accessed 28 January 2016).

Iyer, B. and Gottlieb, R. (2004) 'The four-domain architecture: an approach to support enterprise architecture design', *IBM Systems Journal*, Vol. 43, No. 3, pp.587–597.

Iyer, B., Dreyfus, D. and Gyllstrom, P. (2007) *A Network-based View of Enterprise Architecture*, Pallab Saha (Ed.), p.306, IGI Global, Hershey.

Jefferies, N., Mitchell, C. and Walker, M. (1996) 'Cryptography: policy and algorithms', in Ed Dawson and Jovan Golic´(Eds.): *International Conference Brisbane, Queensland, Australia, July 3–5, 1995 Proceedings*, Chap. A Proposed Architecture for Trusted Third Party Services, pp.98–104, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN: 978-3-540-49363-1, DOI: 10.1007/BFb0032349 [online] http://dx.doi.org/10.1007/BFb0032349 (accessed 27 January 2016).

Kagal, L., Finin, T. and Joshi, A. (2005) *Rei: A Policy Specification Language* [online] http://rei.umbc.edu (accessed 27 January 2016).

Kagal, L. and Abelson, H. (2010) 'Access control is an inadequate framework for privacy protection', in *W3C Privacy Workshop*, pp.1–6.

Kalnis, P., Ghinita, G., Mouratidis, K. and Papadias, D. (2007) 'Pre-venting location-based identity inference in anonymous spatial queries', *Knowledge and Data Engineering, IEEE Transactions on*, Vol. 19, No. 12, pp.1719–1733.

Karjoth, G., Schunter, M. and Waidner, M. (2003) 'Privacy enhancing technologies', in Roger Dingledine and Paul Syverson (Eds.): *Second International Workshop, PET 2002 San Francisco, CA, USA, April 14–15, 2002 Revised Papers*, Chap. Platform for Enterprise Privacy Practices: Privacy-Enabled Management of Customer Data, pp.69–84, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN: 978-3-540-36467-2, DOI: 10.1007/3-540-36467-6_6 [online] http://dx.doi.org/10.1007/3-540-36467-6_6 (accessed 27 January 2016).

KDE (2009) *Social Desktop for KDE* [online] http://techbase.kde.org/Projects/Social-Desktop (accessed 7 January 2016).

KDE (2014) *Nepomuk* [online] https://userbase.kde.org/Nepomuk (accessed 7 January 2016).

Kelley, P.G., Bresee, J., Cranor, L.F. and Reeder, R.W. (2009) 'A nutrition label for privacy', in *Proceedings of the 5th Symposium on Usable Privacy and Security*, ACM, p.4.

KPMG International Cooperative (2010) *Convergence Goes Mainstream: Convenience Edges Out Consumer Concerns Over Privacy and Security* [online] http://www.kpmg.com/CN/en/IssuesAndInsights/ArticlesPublications/Documents/Convergence-Goes-Mainstream-O-201007.pdf (accessed 7 January 2016).

Kudo, M. and Hada, S. (2000) *XML Access Control Language (XACL)* [online] http://xml.coverpages.org/xacl.html (accessed 7 January 2016).

Kumaraguru, P., Cranor, L., Lobo, J. and Calo, S. (2007) 'A survey of privacy policy languages', in *Workshop on Usable IT Security Management (USM 07): Proceedings of the 3rd Symposium on Usable Privacy and Security*, ACM.

LinkedIn Corporation (2011) *LinkedIn* [online] http: //www.linkedin.com (accessed 7 January 2016).

Liu, K., Miklau, G., Pei, J. and Terzi, E. (2010) 'Privacy-aware data mining in information networks', in *KDD 2010 Tutorial*.

Lorea (2010) *Lorea* [online] https://web.archive.org/web/20150624210724/ https://gitorious.org/lorea (accessed 7 January 2016).

Lugano, G. and Saariluoma, P. (2007) 'User modeling 2007', in Cristina Conati, Kathleen McCoy and Georgios Paliouras (Eds.): *11th International Conference, UM 2007, Corfu, Greece, July 25–29, 2007. Proceedings*, Chap. To Share or Not to Share: Supporting the User Decision in Mobile Social Software Applications, pp.440–444, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN: 978-3-540-73078-1, DOI: 10.1007/978-3-540-73078-1_61 [online] http://dx.doi.org/10.1007/978-3-540-73078-1_61 (accessed 27 January 2016).

Markides, B. and Coetzee, M. (2008) 'BlueTrust in a real world', in *Availability, Reliability and Security, 2008. ARES 08. Third International Conference on*, IEEE, pp.440–445.

Mokbel, M.F., Chow, C-Y. and Aref, W.G. (2006) 'The new Casper: query processing for location services without compromising privacy', in *Proceedings of the 32nd International Conference on Very Large Data Bases*, VLDB Endowment, pp.763–774.

Myspace LLC (2014) *Myspace LLC* [online] https://myspace.com (accessed 7 January 2016).

Narayanan, A. and Shmatikov, V. (2010) 'Myths and fallacies of personally identifiable information', *Communications of the ACM*, Vol. 53, No. 6, pp.24–26.

Noll, J., Chowdhury, M.M.R., Kálmán, G. and Gomez, J.M. (2007) 'Semantically supported authentication and privacy in social networks', in *Emerging Security Information, Systems, and Technologies, 2007. SecureWare 2007. The International Conference on*, IEEE, pp.83–88.

OASIS Standard (2005) *eXtensible Access Control Markup Language (XACML) Version 3.0* [online] http://docs.oasis-open.org/ xacml/3.0/xacml-3.0-core-spec-os-en.pdf (accessed 7 January 2016).

Office of the Australian Information Commissioner (1988a) *Information Privacy Principles* [online] https://www.oaic.gov.au/privacy-law/privacy-archive/privacy-resources-archive/information-privacy-principles (accessed 8 January 2016).

Office of the Australian Information Commissioner (1988b) *National Privacy Principles* [online] http://www.oaic.gov.au/ privacy/privacy-act/national-privacy-principles (accessed 8 January).

Opera Software ASA (2011) *Opera Unite* [online] http://help.opera.com/Windows/ 12.10/en/unite.html (accessed 8 January 2016).

Organisation for Economic Co-operation and Development (2002) *OECD Guidelines on the Protection of Privacy and Transborder Flows of Personal Data*, OECD Publishing [online] http://bit.ly/OECDGuildelines (accessed 27 January 2016).

Park, H-a., Hong, J.W., Park, J.H., Zhan, J. and Lee, D.H. (2010) 'Combined authentication-based multilevel access control in mobile application for DailyLifeService', in *IEEE Transactions on Mobile Computing*, Vol. 9, No. 6, pp.824–837 ISSN: 1536-1233 [online] DOI: 10.1109/ TMC.2010.30 (accessed 27 January 2016).

PayPal (1999–2016) *PayPal* [online] https://www.paypal.com/au/webapps/mpp/home (accessed 8 January 2016).

PeerSoN (2016) *PeerSoN: Privacy-Preserving P2P Social Networks* [online] http://www.peerson.net/index.shtml (accessed 8 January 2016).

Peng, W-C., Wang, T-W., Ku, W-S., Xu, I. and Hamilton, J.A. (2008) 'A cloaking algorithm based on spatial networks for location privacy', in *Sensor Networks, Ubiquitous and Trustworthy Computing, 2008. SUTC'08. IEEE International Conference on*, IEEE, pp.90–97.

Preibusch, S. and Beresford, A.R. (2009) 'Privacy-preserving friendship relations for mobile social networking', in *W3C Workshop on the Future of Social Networking*, Citeseer.

Puttaswamy, K.P.N. and Zhao, B.Y. (2010) 'Preserving privacy in location-based mobile social applications', in *Proceedings of the 4th International Conference on Next Generation Mobile Applications, Services and Technologies (NGMAST)*, ACM, pp.1–6.

Raban, D.R., Ricken, S.T., Grandhi, S.A., Laws, N. and Jones, Q. (2009) 'Hello stranger! A study of introductory communication structure and social match success', in *Proceedings of the 42nd Hawaii International Conference on System Sciences, 2009. HICSS'09*, IEEE, pp.1–9.

Roig, A. (2010) 'Privacy and social networks: from data protection to pervasive computing', in *AAAI Spring Symposium Series*, Association for the Advancement of Artificial Intelligence, Palo Alto, California.

Sadeh, N., Hong, J., Cranor, L., Fette, I., Kelley, P., Prabaker, M. and Rao, J. (2009) 'Understanding and capturing people's privacy policies in a mobile social networking application', *Journal of Personal and Ubiquitous Computing*, Vol. 13, No. 6, pp.401–412.

Schulzrinne, H., Tschofenig, H., Morris, J., Cuellar, J., Polk, J. and Rosenberg, J. (2007) *Common Policy: A Document Format for Expressing Privacy Preferences*, DOI: 10.17487/RFC4745 [online] http://www.rfc-editor.org/info/rfc4745 (accessed 12 January 2016).

Security Services Technical Committee (SSTC) (2004) *Security Assertion Markup Language (SAML) 2.0*, The Organization for the Advancement of Structured Information Standards (OASIS) [online] http://xml. coverpages.org/SAML-TechOverviewV20-Draft7874.pdf (accessed 5 January 2016).

Sessions, R. (2007) *A Comparison of the Top Four Enterprise-Architecture Methodologies*, Tech. rep., Microsoft [online] http://msdn.microsoft.com/en-us/library/bb466232.aspx (accessed 8 January 2016).

Shapiro, S.S. (2010) 'Privacy by design: moving from art to practice', *Communications of the ACM*, Vol. 53, No. 6, pp.27–29.

Spiekermann, S. and Cranor, L.F. (2009) 'Engineering Privacy', *IEEE Transactions on Software Engineering*, Vol. 35, No. 1, pp.67–82, ISSN: 0098-5589, DOI: 10.1109/TSE. 2008.88.

StatusNet Inc. (2010) *StatusNet* [online] http://www.gnu.org/software/social/ (accessed 27 January 2016).

Studi, P., Mohomed, I., Balakrishnan, M., Morley Mao, Z., Ramasubramanian, V. and Wobber, T. (2010) *The Cloud Is the Router: Enabling Bandwidth-Efficient and Privacy-Aware Mobile Applications with Contrail*, Tech. rep.

Tan, X., Qin, L., Kim, Y. and Hsu, J. (2012) 'Impact of privacy concern in social networking web sites', *Internet Research*, Vol. 22, No. 2, pp.211–233, DOI: 10.1108/10662241211214575, eprint: http://dx.doi.org/10.1108/10662241211214575 [online] http://dx.doi.org/10.1108/10662241211214575 (accessed 27 January 2016).

The Internet Engineering Task Force (2014) *Extensible Messaging and Presence Protocol* [online] http://xmpp.org/ (accessed 8 January 2016).

The Open Group (2013) *The Open Group Architectural Framework (TOGAF)* Tech. rep., The Open Group [online] http://www.opengroup.org/togaf/ (accessed 8 January 2016).

Twitter Inc. (2014) *Twitter* [online] https://twitter.com/ (accessed 8 January 2016).

United States Executive Office (2012) *Federal Enterprise Architecture* [online] http://www.whitehouse.gov/sites/default/files/omb/assets/egov_docs/common_approach_to_federal_ea.pdf (accessed 8 January 2016).

United States Federal Trade Commission (1973) *Fair Information Principles* [online] http://www.dhs.gov/xlibrary/assets/ privacy/privacy_policyguide_2008-01.pdf (accessed 8 January 2016).

Vodafone Group (2010) *Onesocialweb* [online] https://www.crunchbase.com/organization/onesocialweb#/entity (accessed 8 January 2016).

Von Arb, M., Bader, M., Kuhn, M. and Wattenhofer, R. (2008) 'Veneta: serverless friend-of-friend detection in mobile social networking', in *Proceedings of the IEEE International Conference on Wireless and Mobile Computing Networking and Communications (WIMOB '08)*, IEEE, Avignon, pp.184–189.

Wallbridge, R. (2009) 'How safe is Your Facebook Profile? Privacy issues of online social networks', *The ANU Undergraduate Research Journal One*, p.85 [online] https://eview.anu.edu.au/anuuj/vol1_09/pdf/whole_book.pdf#page=101 (accessed 27 January 2016).

Weiss, S. (2009) 'Privacy threat model for data portability in social network applications', in *International Journal of Information Management*, Vol. 29, No. 4, pp.249–254, ISSN: 0268-4012, DOI: http://dx.doi.org/10.1016/j.ijinfomgt.2009.03.007 [online] http://www.sciencedirect.com/science/article/pii/S0268401209000474 (accessed 27 January 2016).

Wikimedia Foundation Inc. (2015) *Comparison of Software and Protocols for Distributed Social Networking* [online] https://en.wikipedia.org/wiki/Comparison_of_software_and_protocols_for_distributed_social_networking (accessed 4 January 2016).

Williams, J.B. and Weber-Jahnke, J.H. (2010) 'Social networks for health care: addressing regulatory gaps with privacy-by-design', in *Privacy Security and Trust (PST), 2010 Eighth Annual International Conference on*, pp.134–143.

Wishart, R., Corapi, D., Madhavapeddy, A. and Sloman, M. (2010) 'Privacy butler: a personal privacy rights manager for online presence', in *Proceedings of the 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*.

Xu, T. and Cai, Y. (2007) 'Location anonymity in continuous location-based services', in *Proceedings of the 15th Annual ACM International Symposium on Advances in Geographic Information Systems*, ACM, p.39.

Xu, T. and Cai, Y. (2008) 'Exploring historical location data for anonymity preservation in location-based services', in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, DOI: 10.1109/INFOCOM.2008.103.

Yahoo! Inc. (2014) *Flickr* [online] https://www.flickr.com/ (accessed 29 January 2016).

YouTube LLC (2014) *Youtube* [online] https://www.youtube.com/ (accessed 20 June 2014).

Zachman, J.A. (1987) 'A framework for information systems architecture', *IBM Systems Journal*, Vol. 26, No. 3, pp.276–292, ISSN: 0018-8670, DOI: 10.1147/sj.263.0276.

Zachman, J.A. (2008) *The Zachman Framework* [online] http://zachman.com/about-the-zachman-framework (accessed 8 January 2016).

Zero-Knowledge Systems Inc. (2004) *Privacy Rights Markup Language (PRML) Specification* [online] http://xml.coverpages.org/ZKSMotions.html (accessed 8 January 2016).

Notes

1 https://en.wikipedia.org/wiki/Comparison_of_software_and_protocols_for_distributed_social_networking
2 https://github.com/ijoey/6d
3 https://github.com/tav/ampify
4 https://github.com/anahitasocial/anahita
5 https://github.com/appleseedproj
6 http://buddycloud.com
7 http://www.cunity.net
8 https://diasporafoundation.org
9 https://diso-project.org
10 http://www.colm.net/files/dsnp/dsnp-overview.pdf
11 http://duuit.com
12 http://www.friend2friend.com
13 http://friendica.com
14 http://www.gnu.org/software/social/
15 https://jappix.org
16 https://www.know.ee/help?locale=en
17 https://code.google.com/p/kopal
18 http://kune.ourproject.org
19 https://github.com/philcryer/lipsync
20 http://libertree.org
21 https://lorea.org/
22 http://p2pfoundation.net/Lorea
23 https://github.com/edhelas/movim
24 http://mobisocial.stanford.edu/papers/hotpets11.pdf
25 http://newebe.org
26 https://code.google.com/p/noserub
27 http://andrewrondeau.com/ObjectCloud
28 https://www.crunchbase.com/organization/onesocialweb

29 http://openautonomy.com
30 http://ods.openlinksw.com/wiki/ODS
31 https://github.com/voitto/openmicroblogger
32 http://projectdanube.org
33 http://pde.cc/tags/project-nori
34 http://www.psyced.org
35 http://pump.io
36 https://github.com/redmatrix/redmatrix
37 http://retroshare.sourceforge.net
38 http://salut-a-toi.org
39 https://github.com/smob/smob
40 https://social-igniter.com
41 http://socialriver.org
42 http://sourceforge.net/projects/socialze
43 http://sourceforge.net/projects/foomor-socknet
44 https://github.com/Bombe/Sone
45 http://sparkleshare.org
46 https://tent.io
47 http://www.transmediale.de/thimbl-decentralized-microblogging
48 http://twister.net.co
49 http://www.w3.org/2005/Incubator/federatedsocialweb/wiki/WeeStit
50 https://www.anonymizer.com/