Inter-Social-Networking: Accounting for Multiple Identities

Dominic Price[1], Derek McAuley[1], Richard Mortier[2], Chris Greenhalgh[1], Michael Brown[1], Spyros Angelopoulos[1]

[1] Horizon Digital Economy Research, University of Nottingham, UK
{firstname.lastname}@nottingham.ac.uk
[2] Systems Research Group, Cambridge University Computer Laboratory, UK
richard.mortier@cl.cam.ac.uk

Abstract. We argue that the current approaches to online social networking give rise to numerous challenges regarding the management of the multiple facets of people's digital identities within and around social networking sites (SNS). We propose an architecture for enabling people to better manage their SNS identities that is informed by the way the core Internet protocols developed to support interoperation of proprietary network protocols, and based on the idea of Separation of Concerns [1]. This does not require modification of existing services but is predicated on providing a connecting layer over them, both as a mechanism to address problems of privacy and identity, and to create opportunities to open up online social networking to a much richer set of possible interactions and applications.

Keywords: social-networking · privacy · identity · infrastructure · architecture

1 Introduction

Online social network sites such as Twitter, Facebook and Google+ have become enormously popular over recent years, for example; Facebook now claims to have over 1 billion active accounts.[1] This popularity has been driven by many factors: the desire we have as people to communicate and to belong; the convenience that is provided by offloading contact management to external services; and the entertainment value of keeping up with our friends, family and acquaintances. Their popularity has led to these, and related, services becoming synonymous with social networking, but this popularity masks a number of problems with their implementation and structure.

In real life, social networking is far more broadly defined: as social beings we are well known to benefit from involvement and participation in groups [2]. We participate in multiple, often overlapping, social networks simultaneously rather than one all-encompassing social network as Facebook, Google+, etc. aspire to be. The tension between our many online identities and the desire of companies such as Facebook and Google to force us to present just a single identity through their 'real name' policies

[1] http://news.cnet.com/8301-1023_3-57525797-93

[3] is an example of the evidence that our lives are far more complex than that which is captured by these services.

Observing that at least part of the problem is the centralized nature of these commercial services, some have proposed decentralized equivalents, e.g., Diaspora.[2] Decentralization certainly mitigates many of the problems of the large centralized social networks, e.g., resilience, need for personal control, support for multiple identities, but it is not alone sufficient to address the problems raised in online social networking (discussed in section 2).

Our position is informed by observation of the development of computer networking in the 1970s and 1980s: the industry moved from many different, mutually incompatible, proprietary networking standards towards a single common interoperation protocol now recognize as the Internet Protocol (IP). We advocate a similar move in the world of online social networking, both to address problems of privacy and identity, but also to open up social networking to a much richer set of possible interactions and applications.

By analogy with the development of the Internet's TCP/IP protocol suite (discussed in section 3), we argue that social networking requires a new architecture that is sufficiently flexible to encompass the very broad range of social network interactions in which we participate (discussed in section 4). We present and discuss a proposal for such an architecture, alongside initial exploratory prototype development and experiments we are carrying out into its implementation (discussed in section 5).

2 One Size Does Not Fit All

Notwithstanding the efforts of a handful of corporations to become pre-eminent in global social networking, notably Facebook and Google, it seems clear that need for a richer set of services will remain. Moving outside nations where English is commonly spoken, whether as first or second language, we find a rich set of social media services. For example, use of VK is widespread in Russian speaking countries; after having been created in the US, Orkut is now 59% Brazilian; and in China, RenRen and Sina Weibo are used instead of Facebook and Twitter.

However, many other online communities also behave as social networks in terms of the interactions they support (commenting, following, sharing). For example, Amazon reviews, blog-specific commentator communities, personal and community YouTube channels, Github source code repositories, and even non-public Enterprise internal communications. Indeed, companies have been formed to support this broader definition of more specific communities. For example, Ning[3] supports over 2 million communities, ranging in size from just tens to over a million members, within which you can either reuse an existing identity (Facebook, Google+, Yahoo!) or cre-

[2] http://diasporaproject.org/

[3] http://ning.com, acquired by Glam Media in 2011.

ate a fresh one. Examining the enterprise sector we find, for example, Yammer[4] and Jabber.[5]

As this plethora of social networks suggests, many (if not most) of us have multiple online identities through which we actively manage our social interactions. Often these identities are anonymous, and many of us would suffer embarrassment, loss or worse if all these identities were publicly linked. Many reasons why we choose to explicitly manage overlap among our social networks, even keeping some networks completely distinct from others, are completely normal and not in the least clandestine. For example, teenagers wishing to discuss sensitive health matters in online fora [4], employees complaining about treatment at work [5], or those engaged in political commentary in uncomfortable or dangerous situations [6].

Attempts by the major services to support this richness in our social networks via access control mechanisms, e.g., Facebook lists, Google+ Circles, have proved largely inadequate. Typically whilst users understand how these mechanisms work the cognitive effort required for creation and maintenance results in either their mis- or non-use. Furthermore, collating all of one's social interactions and data into a single service gives rise to serious risks such as identity theft.

Fully decentralized systems such as Diaspora have seen some success, but are still very much under construction and do not address the entire problem. Other decentralized versions of common services include status.net for microblogging (centralized equivalent: Twitter), wordpress.org for blogging (centralized equivalent: Wordpress.com), and use of the git repository management system and its built-in webserver for collaboration around code (centralized equivalent: Github). To take Diaspora as one of the more mature examples, it supports asymmetric sharing, and federation among Diaspora pods, whether community or individual, enabling much greater choice over who is trusted.

In all these cases, whether decentralized or not, the lack of proper boundaries between such groupings permits inappropriate linking and unexpected leakage of content and relationships between them and to the outside world [7, 8, 9, 10, 11]. Methods to detect and resolve privacy conflicts [12], and more generally, to limit and monitor information released online [13] are lacking.

The issues we note above are not completely new: some in the W3C have previously noted similar issues concerning uniformity of addressing and access around cloud storage,[6] for example. However, these and other Linked Data approaches to managing our online identities, tend to make presumptions of constituent data being public, and fail to properly address questions such as selective sharing of and delegation of access to data, and the need for an 'app ecosystem' rather than a focus on human consumption of data.[7]

---

4 http://yammer.com/, acquired by Microsoft in 2012.

5 http://www.cisco.com/web/products/voice/jabber.html, acquired by Cisco in 2008.

6 http://www.w3.org/DesignIssues/CloudStorage.html

7 http://www.ted.com/talks/tim_berners_lee_on_the_next_web.html

3 Your Grandfather's Internet

The essence of our argument is that current online social network platforms are ultimately limited, for one of two reasons. First, whether centralized or decentralized they are use-specific and to date have been designed with service specific APIs, limiting the scope of the applications that can be built[8] and requiring that any specific application be implemented afresh for each platform. Second, the terms and conditions commonly prohibit the use of multiple accounts: to express multiple identities we must use several services. Even for those services that permit – or at least, do not prohibit – multiple identities, the possibility of correlating accounts through means as simple as IP access address requires trusting (at least!) the provider to not do this, limiting privacy.

Our approach to this problem develops by analogy[9] to the development of the ARPANET into the CATENET and thus the Internet, and so we next sketch this historic development.

Proposed in the late 1960 and first implemented in 1969, the ARPANET was one of the first operational packet-switched networks.[10] A key part of the early ARPANET was the Host-to-Host protocol, known as the Network Control Protocol (NCP, 1970) [15], that provided connectivity and flow control between processes running on different ARPANET- connected hosts. In 1974 Kahn and Cerf presented a protocol for interconnecting distinct packet networks [16], separating the notions of host-host data transfer and inter-network communication via a sequence of gateways.

In 1977, Jon Postel introduced IEN 2 [17] as follows "The position taken here is that internetwork communication should be view as having two components: the hop by hop relaying of a message, and the end to end control of the conversation." and subsequently wrote "We are screwing up in our design of internet protocols by violating the principle of layering.". In IEN 2 he proposes the split of TCP into IP (known therein as "the Internet Hop Protocol") and TCP (known therein as "the Internet Host Protocol"), and subsequently as IP and TCP. Following the RFC process, the implementation of this split is described in RFC801 [18], with the final "flag day" (January 1st, 1983) when the ARPANET ceased to support NCP, switching over completely to IP/TCP.

Building on previous work by Pouzin [19], Cerf set out the "Catenet[11] strategy for internetworking" [20] in 1978. This, and other, elements were later elaborated into the Internet architecture by Clark [21]. Clearly a great deal of related work took place at the time, and things have developed enormously since then. However, the core of the Internet architecture is this separation between a network layer provided by IP and a transport layer provided by TCP. The former is responsible for addressing and transferring data between hosts (implicitly identified with network interfaces), and the latter for transferring data between processes.

---

8 Never mind the limitations on use placed on many of the APIs.

9 With awareness of the perils of this approach [14].

10 http://www.internetsociety.org/internet/what-internet/history-internet/brief-history-internet

11 From 'concatenated networks'.

This architecture had, ultimately, enormous commercial impact. The simplicity of the IP layer meant that it could easily be ported to run over almost any underlying link layer technology [22]. Over the course of the 1970s, 80s and 90s this led to the diminishing in importance of proprietary local area network technology from vendors such as IBM, DEC and Xerox in favor of support for IP. On the flip side, software developers could cease caring what particular flavor of network they were operating over, and simply assume that IP, and their choice of transport protocol (e.g., TCP, UDP), were available. The result was a steady explosion in the use of the network, reaching back to email in 1972, but continuing with other applications such as FTP, Gopher, the web, BitTorrent, YouTube, and on into the present day and foreseeable future.

4 Refactoring Social Networks

It is easy to see the inherent weakness of the dominant centralized social networks such as Facebook: by imposing a one-size-fits-all model on social interaction, while they (the social networks) may satisfy some of the needs of a large number of people, they will never be able to satisfy all the needs of all the people. In particular, their centralized cloud-hosted nature means that they very poorly, if at all, support our need for multiple online identities where we are in control of the linking of those identities [3]. Referring to our sketch of the Internet's evolution, we see commercial systems like Facebook as analogous to proprietary networking solutions: while initially successful and certainly satisfactory for many customers, they were too restrictive to enable the explosion of use that the Internet subsequently saw. Only by interposing a simple interconnection layer such as IP could the complexity of development for these proprietary networks be contained, and their utility accessed.

Decentralized approaches such as Diaspora address that weakness to some extent, but a more subtle – but still serious – problem remains. By baking the data types handled by the system into the data exchange protocols, users must either cast the data they wish to exchange into the formats supported, or install expensive, brittle and bug-prone gateways to interconnect different networks (cf. "Relay Service" [17]). Again, comparing to our sketch of the Internet' evolution, this is analogous to the mistake noted in the early development of the Internet protocols. Only by separating concerns between IP and TCP (and other transport protocols subsequently) could the combination of absolute flexibility and a simple, uniform protocol interface be provided.

Finally, a key requirement for providing the levels of access control, communication privacy and (where desired) authenticated identity required by such disparate and personal interaction is the ability to securely and coherently generate, manage and distribute secrets. Only by providing consistent mechanisms for deriving and distributing appropriate public key material can we begin to meet the complex, multi-faceted identity needs of real life.

In short, without a simple way to interconnect and manage our identities on different social networks, we will not see the same explosion in creativity that the Internet gave rise to.

We close this discussion of the merits of refactoring currently popular social network services with a key observation concerning email. Since the early days of the ARPANET (ca. 1972), Internet users have commonly used several email addresses, e.g., to distinguish personal and university/corporate communication. Indeed, today's Internet users are often forced to have several email addresses, e.g., one in the cloud that provides some longevity, and one forced upon them by their ISP for ISP to consumer communication.

As a result, we have designed the tools – primarily email clients – to understand and manage this. Importantly, the only point at which these our many email identities must exist together is within those clients in the private context of our personal devices such as mobile phone, tablet and personal computer.[12]

5 An Inter-Social Network

We next elaborate on the technical implications of such a separation of concerns. We believe that the key features for an inter-SNS layer are: transport-independent addressing, format standardization for referring to data distributed through a particular social network, and flexible – but standardized – support for use of asymmetric encryption for per-service and per-recipient authentication and privacy. These features can be achieved through:

Loose binding of identities. Where a single person has multiple online identities, the linking of these identities will only be performed in client software based on out-of-band information; identities need not explicitly contain any information that can be used to link them together. Instead linking must rely on information provided by the person to whom the identities refer, or the knowledge of the person who has a social network connection to the first person. This will allow messages being produced by the same author to be identified whilst also protecting the identities of the author.

Semi-structured data. By defining and making available schema details for messages, and providing enough information within the message (the semi-structure) to determine the scheme, the online social network service provider is at liberty to structure their messages as they see fit. In some cases a message might be nothing more than a string of specified length, or an image; in others, a message might have very rich structure, with extensive metadata in addition to the raw content.

Asymmetry & Authentication. Schema can be defined that support part or all of a message being encrypted, enabling privacy and authentication. Contacts would be

[12] Some folk may choose to configure a single cloud-hosted email service to fetch mail from all their accounts for simplicity, but this is a personal choice and by doing so they risk making confidential information available for data-mining by cloud-providers.

required to associate trust relationships through out-of-band mechanisms such as OAuth or face-to-face interaction.

5.1 Discussion

Support for legacy social networks is straightforward to achieve, much as IP was provided over legacy proprietary networks. From a technical point-of-view, one could provide service shims[13] that implement the above abstraction over existing proprietary APIs. As noted above however the brittle nature of these APIs would necessitate constant updating of these shims to track the changes in the APIs, the technical challenge is therefore only one piece of the puzzle. From a regulatory point-of-view it looks increasingly likely that moves such as upcoming EU data protection regulation and the UK's midata[14] initiative will enshrine a right for each of us to extract all the data by and about us held by a social network, enabling us easily to move away from legacy services. Applications can then be written that make use of the simplicity and transparency of the new API (as specified by these regulations), enabling much smoother and richer social integration in our online interactions rather than the current state where we are limited to either using a social network identity with a third party service (and the privacy infringement that entails) or simple (re-)posting of data on pre-specified services.

5.2 Current Implementation

We are currently working towards the creation of a prototype social networking platform that will allow us to experiment with the key features of; *loose binding of identities, semi-structured data* and *asymmetry and authentication* as described above. Whilst there are many aspects to social networking, including identity and relationships (e.g., Twitter followers, Facebook friends), we have chosen messaging as our starting point and are developing prototypes in order to experiment and study with message structure and message transport; our initial goal being the development of a generic client for aggregating content across multiple social networks. **Fig. 1.** shows a high-level diagram of the prototype architecture.

The *shim layer* is responsible for fetching messages (Facebook feed, Twitter timeline) and contacts from existing social networks (we have currently implemented shims for Facebook and Twitter) and posting messages to these networks. It is also responsible for conversion of messages and contacts from the network specific formats to the format used by the prototype (*semi-structured data*). The *storage* format we are experimenting with for storing messages in the Internet Message Format [23] (commonly referred to as email). Through MIME [24], an email body is able to store a wide variety of content, especially content typically associated with social network messages (text, images, video, etc.) making it an appropriate selection for message

[13] http://en.wikipedia.org/wiki/Shim_%28computing%29

[14] https://www.gov.uk/government/policies/providing-better-information-and-protection-for-consumers/supporting-pages/personal-data

storage. Using this format also allows us to re-use existing email clients as *clients* for the prototype (e.g., we can use the Thunderbird[15] email client to read and write messages).

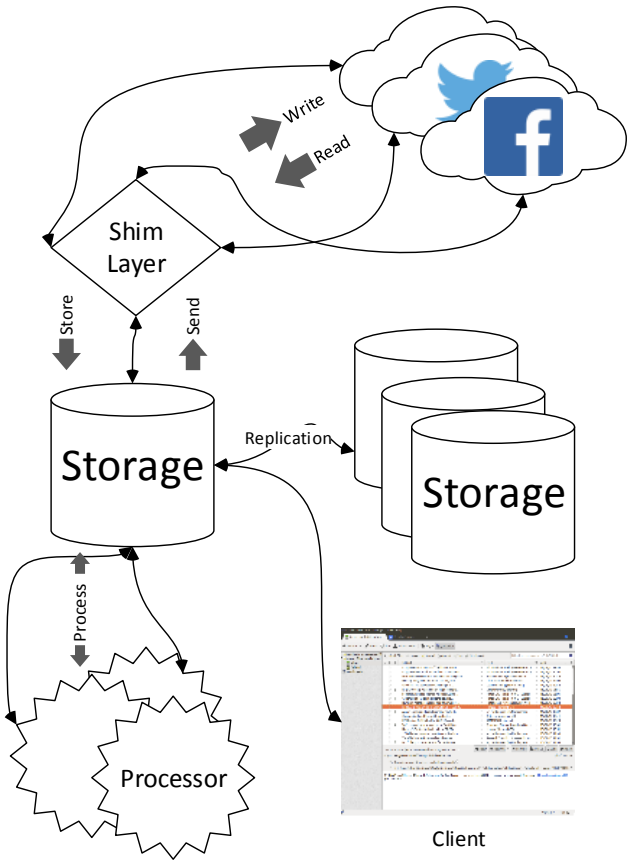

Fig. 1. Prototype platform architecture

Multiple social network accounts can be registered with the shim layer, where each account is treated as a separate identity. The platform is the only place where these identities are explicitly associated with each other (*loose binding of identities*). A user of the platform is able to aggregate content from their different accounts/identities but they are the only person who can view the aggregate, when sending messages they can explicitly control the account/identity (or multiples thereof) that is (are) used to transmit the message.

The *processor* layer is a queue through which incoming and outgoing messages are routed. Each processor is a small independent application that performs actions on

[15] https://www.mozilla.org/en-GB/thunderbird/

each message as the message passes through the queue. One example processor that we are developing is to provide encryption for outgoing messages where the recipient is known to accept encrypted messages. Other suggested uses for the processor layer are to provide cross-social network search or integrated spam filtering.

Finally, *replication* of the storage layer across multiple locations allows redundancy and for clients to access messages from multiple independent locations, e.g., a mobile device could maintain a copy of the store so that messages can be accessed in periods of no-connectivity. In addition, replication will allow processors to run in the most appropriate environment, e.g., a processor intensive processor could run in the cloud on a high-performance device rather than a mobile device where computational power is at a premium.

5.3 Evaluation

At this present time, the prototype has not yet reached a point of development at which we can begin to experiment with it and evaluate it fully. We are able however to provide the following evaluation based on its current status. In particular we can begin to evaluate the implications of transferring the responsibility of message storage from social network providers to the individual. Taking the specific example of Twitter as a social network service provider, message storage is handled by Twitter and is thus of zero-cost to an individual user. Within our proposed system, storage for messages must be provided by the user the space requirements for which will only grow over time and thus incur a cost (options such as free online storage providers are an option there is however still an administrative cost involved in setting up and maintaining these). An analysis of the authors' Twitter streams when mapped to a MIME email message reveals an average size per message of 2 kilobytes (increasing significantly if the Tweet has an image attached). Whilst usage between users will vary greatly in terms of numbers of messages received we can begin to use this figure as a baseline to calculate storage costs. For example, a user receiving an average of 100 Tweets per day will require storage for approximately 71 megabytes within a year. Alone, a fairly trivial amount of storage but once we start factoring in other social networks, particularly media rich networks such as Flickr, this figure could increase significantly.

6 Conclusions

In this paper we have elaborated on three serious problems that we perceive with the current state of online social networking, from a systems/networking point-of-view. As a result, we believe it is necessary to revisit the ways that we have been architecting and building online social networking platforms, to provide a cleaner separation of layers. This will enable greater flexibility, creativity and utility in the exploitation of our social graphs, while also providing us with greater control over that exploitation. We believe that doing so will open social networking up to richer

application development, and so will enable the same kind of explosion in use that the Internet caused with computer networking.

Acknowledgements. This work is supported by Horizon Digital Economy Research, RCUK grant EP/G065802/1; and by CREATe, the Centre for Copyright and New Business Models, RCUK grant AH/K000179/1. Packages and source are available under open source licenses at github.com/CREATe-centre/.

7 References

1. W.L. Hürsch, & C.V. Lopes. Separation of concerns. Technical report, 1995. North Eastern University.
2. A. Portes. Social capital: Its origins and applications in modern sociology. Annual Review of Sociology, 24(1):1–24, 1998.
3. L. Edwards and D. McAuley. What's in a name? Real name policies and social networks. In Proceedings of 1st International Workshop on Internet Science and Web Science Synergies (INETWEBSCI), Paris, France, May 1 2013.
4. M. van der Velden and K. El Emam. "not all my friends need to know": a qualitative study of teenage patients, privacy, and social media. Journal of the American Medical Informatics Association, 2012.
5. C. N. O'Brien. The top ten NLRB cases on Facebook firings and employer social media policies. Oregon Law Review, 92(2), Winter 2014. http://ssrn.com/abstract=2277900.
6. A. M. Attia, N. Aziz, B. Friedman, and M. F.Elhusseiny. Commentary: The impact of social networking tools on political change in Egypt's "revolution 2.0". Electronic Commerce Research and Applications, 10(4):369–374, 2011.
7. M. Madejski, M. L. Johnson, and S. M. Bellovin. The failure of online social network privacy settings. Technical Report CUCS-010-11, Columbia University Computer Science, Columbia University, New York, USA, 2011.
8. F. K. Ozenc and S. D. Farnham. Life "modes" in social media. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '11, pages 561–570, Vancouver, BC, Canada, 2011.
9. E. Zheleva and L. Getoor. To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles. In Proceedings of the 18th international conference on World Wide Web, WWW '09, pages 531–540, Madrid, Spain, 2009.
10. S. Le Blond, C. Zhang, A. Legout, K. Ross, and W. Dabbous. I know where you are and what you are sharing: exploiting p2p communications to invade users' privacy. In Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference, IMC '11, pages 45–60, Berlin, Germany, 2011.
11. J. a. P. Pesce, D. L. Casas, G. Rauber, and V. Almeida. Privacy attacks in social media using photo tagging networks: a case study with Facebook. In Proceedings of the 1st Workshop on Privacy and Security in Online Social Media, PSOSM '12, pages 4:1–4:8, Lyon, France, 2012.
12. H. Hu, G.-J. Ahn, and J. Jorgensen. Detecting and resolving privacy conflicts for collaborative data sharing in online social networks. In Proceedings of the 27th Annual Computer Security Applications Conference, ACSAC '11, pages 103–112, Orlando, Florida, 2011. ACM.

13. C. E. Gates. Access control requirements for web 2.0 security and privacy. In Proc. of Workshop on Web 2.0 Security & Privacy (W2SP 2007, 2007. [15] F. Halasz and T. P. Moran. Analogy considered
14. F. Halasz and T. P. Moran. Analogy considered harmful. In Proceedings of the 1982 Conference on Human Factors in Computing Systems, CHI ’82, pages 383–386, Gaithersburg, Maryland, USA, 1982.
15. S. Crocker. Protocol Notes. RFC 36, IETF, March 1970.
16. V. G. Cerf and R. E. Kahn. A protocol for packet network interconnection. IEEE Transactions on Communications, 22(5):637–648, May 1974
17. J. Postel. 2.3.3.2 Comments on Internet Protocol and TCP. IEN 2, ISI, August 15 1977.
18. J. Postel. NCP/TCP transition plan. RFC 801, IETF, November 1981
19. L. Pouzin. A proposal for interconnecting packet switching networks. In Proceedings of EUROCOMP, pages 1023–1036, Bronel University, May 1974.
20. V. Cerf. The catenet model for internetworking. IEN 48, DARPA/IPTO, July 1978.
21. D. Clark. The design philosophy of the DARPA Internet protocols. In Proceedings of ACM SIGCOMM 1988, pages 106–114, Stanford, California, USA, 1988.
22. D. Waitzman. Standard for the transmission of IP datagrams on avian carriers. RFC 1149, IETF, April 1990.
23. P. Resnick and Ed. Internet Message Format. RFC 2822, IETF, April 2001.
24. N. Freed and N. Borenstein. Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies. RFC 2045, IETF, November 1996.