Computer Networks 75 (2014) 437–452

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

A survey on decentralized Online Social Networks

Thomas Paul [a,*], Antonino Famulari [b], Thorsten Strufe [c]

[a] Technische Universität Darmstadt, Hochschulstrasse 10, 64380 Darmstadt, Germany
[b] Télécom ParisTech, INFRES, 23, Avenue d'Italie, 75013 Paris, France
[c] Technische Universität Dresden, Nöthnitzer Straße 46, 01187 Dresden, Germany

ARTICLE INFO

Article history:
Received 20 January 2014
Received in revised form 15 September 2014
Accepted 1 October 2014
Available online 12 October 2014

Keywords:
Online Social Network (OSN)
Distributed Online Social Network (DOSN)
Privacy

ABSTRACT

Because of growing popularity of Online Social Networks (OSNs) and huge amount of sensitive shared data, preserving privacy is becoming a major issue for OSN users. While most OSNs rely on a centralized architecture, with an omnipotent Service Provider, several decentralized architectures have recently been proposed for decentralized OSNs (DOSNs). In this work, we present a survey of existing proposals. We propose a classification of previous work under two dimensions: (i) types of approaches with respect to resource provisioning devices and (ii) adopted strategies for three main technical issues for DOSN (decentralizing storage of content, access control and interaction/signaling). We point out advantages and limitations of each approach and conclude with a discussion on the impact of DOSNs on users, OSN providers and other stakeholders.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

Online Social Networks (OSNs) are one of the most popular services on the Web with more than one billion users.[1] Yet, OSNs are not just tremendously popular; they have also changed the way we interact with the Web and its resources: OSNs allow non-expert users to create and maintain a personal space in the Web in a simple way. OSNs evolved in time, providing several communications and sharing facilities which cause users to share huge quantities of personal information [22] over them.

Nevertheless, users of today's popular OSNs suffer undesired side effects resulting from a centralized architecture of OSNs in which one provider has the power that is accompanied with being the operator of the system. These side effects include: the necessity for a high degree of trust in the OSN provider, censorship issues and privacy concerns.

Economic pressure to earn money due to provider-side infrastructure and maintenance costs and the provider's legitimate profit interests lead to strong incentives for OSN providers to monetize user data far beyond the user's sharing interests [20]. But users need to trust the latter not to misuse the power, accompanied with being the operator of the system, as well as to be able to protect the system against both attackers from outside and from inside the provider's organization itself. The existence of a powerful system operator combined with monetization incentives cause privacy concerns [27]. Furthermore, different types of censorship occur in today's OSNs: the content-specific censorship with respect to different rules and traditions in different countries [1,2] as well as person-specific censorship which means disallowing subsets of the population to access the network (e.g. in Syria today [3]).

However, the importance of OSNs for the daily inter-person communication puts the OSN providers in a position of being gate keepers to parts of the social life of their users. Due to this dependency, users strongly tend to accept the mentioned side effects and even disadvantageous terms of usage, since the OSN providers may exclude users from the OSNs and subsequently from parts of their social contacts. Authors of decentralized OSN (DOSN)

* Corresponding author.
[1] http://allfacebook.de/userdata/ – accessed on 16th of January 2014.

http://dx.doi.org/10.1016/j.comnet.2014.10.005
1389-1286/© 2014 Elsevier B.V. All rights reserved.

approaches aim to abolish OSN providers and the side-effects of centralized OSNs by creating decentralized systems, providing the social networking functionality.

Approaches for encrypting content in centralized OSNs (e.g. [27]) may mitigate content censorship and communication confidentiality concerns but they still allow the OSN provider to observe communication patterns. Individuals can still be excluded from the system to conduct censorship. Those encryption approaches also raise the question whether the business models of today's OSNs still work and allow the providers to make sufficient infrastructure available. We thus argue that decentralization is the best available concept to address the trust, the privacy and the censorship issues.

Many kinds of DOSNs have been proposed by several authors. Nevertheless, the idea of distributing OSNs has not been widely adopted. Beside Diaspora,[2] none of the DOSNs has a denotative user basis. In contrast to the authors of many DOSNs, Narayanan et al. doubt in [34] that decentralizing OSNs is a feasible way to build social networking services. We argue that distributing OSNs is a worthwhile idea and aim to help the DOSN community with this survey by elaborating and evaluating what has been suggested in the field of DOSN.

In the remainder of this survey, we define the terms, which are elementary for this article in Section 2. We state our requirements and adversary models (Section 3) and introduce a DOSN architecture model to explain the design space (Section 4) as well as the design decisions that have to be made in case of creating DOSN architectures (Section 5). These design decisions determine the properties of DOSNs. Thus, they are the basis for our classification in Table 1. In Section 6, we discuss both: the consequences caused by the design decisions as well as the properties of DOSN classes (with respect to our classification). Furthermore we list all approaches which fit in the shape of our DOSN definition, discuss the features of the different DOSN approaches and provide a publication time line to illuminate the publication history (Section 7). Several ideas to improve aspects in the field DOSNs have been published without suggesting a complete new architecture. Since we consider them to be important contributions, we introduce a selection of DOSN related approaches in Section 9. Finally, we elaborate the impact of decentralization on different OSN affiliates and summarize and conclude our article.

2. Definitions

This Section defines specific terms used in this article: Section 2.1 contains elementary terms, Section 2.2 defines different types of decentralization as well as terms that are closely related to decentralization and Section 2.3 specifies DOSN components.

2.1. Elementary terms

We use the term *Personal identifiable information (PII)* as it has been introduced by the Data Protection Directive 95/46/EC of the European Parliament. PII is "any information relating to a [...] natural person [...] who can be identified, directly or indirectly, in particular by reference [...] to one or more factors specific to his physical, physiological, mental, economic, cultural, or social identity". PII can consist of *content*, e.g. pictures and messages, as well as all types of *incidental data* which can be derived from e.g. technical properties (e.g. size) of content or communication parameters.

Authorization is a mechanism to decide about legitimization based on previously defined rules. *Access control* denotes the action to allow legitimized subjects to access content and to prohibit unauthorized access. According to "privacy as control" in [16], *privacy* in our context means the effectiveness of users to be able to restrict access to information that the user is responsible for (as a producer of the message). The effectiveness can be affected by technical and social (e.g. social engineering) interferences.

An *Online Social Network* (OSN) in this article is "an online platform that (1) provides services for a user to build a public profile and to explicitly declare the connection between his or her profile with those of the other users; (2) enables a user to share information and content with the chosen users or public" [37].

Since each OSN provides different functionality, we decided to distinguish *basic functionality* which needs to be part of the system to be considered being an OSN and the *extended functionality*, which is not qualifying but extends the service in a specific way.

Derived from the definition of OSN, *basic functionality* consists of:

- *Profile management:* creating, maintaining and deleting of user profiles, which subsequently includes authorization mechanisms for profile attributes.
- *Relationship handling:* establishing and removing new relationship declarations.
- *Interaction:* direct interactions (internal messaging system – 1:1) and indirect interactions by sharing content (1:*n*).

The *extended functionality*, we define for this article, is a set of features which some of today's OSNs provide. The reason for mentioning them in this definition is that we consider these to be important pieces, contributing to the attractiveness and popularity of the OSN platforms. We take the following into account:

- An *API*, allowing third party applications to run on the OSN platform.
- A *search* function to find other users of the OSN.
- A *recommender system* that recommends users to become friends or content to be consumed.
- A *social network connector*, bridging different social networks.

2.2. OSN decentralization definitions

Decentralization has more than one dimension in the field of OSN. We distinguish between *technical decentralization* (resource distribution) which means that parts of

[2] https://joindiaspora.com/ – accessed on 14th of January 2014.

Table 1
DOSN approaches; (D = distributed, FD = fully distributed, BE = broadcast encryption, OOB = out of band).

Arch.	Ref	Degree	Storage		AC		Interact Mech.		Comments
			Peers	Server	ACLs	Encr.	Centr.	Dec.	
P2P-OSN	PeerSoN	FD	Previous download	–	–	– PKI	–	Direct + DHT	Support for direct interactions (also with no Internet access)
	Safebook	FD	Trusted friends	–	–	PKI	–	DHT	Anonymity of interactions via encryption and recursive hop-by-hop routing
	LifeSocial. KOM	FD	DHT	–	–	BE	–	Plugins	Interactions based on external applications (plugins)
	LotusNet	FD	DHT	–	–	PKI	–	DHT	Based on Likir
	DECENT	FD	Random nodes in a DHT	–	–	ABE	–	DHT	Social network functionality on top of EASIER
	Cachet	FD	Random nodes in a DHT	–	–	ABE	–	DHT	Performance improvement of DECENT
F-OSN	SoNet	FD	–	Active	Servers	OOB or SMP	–	XMPP	XMPP-like architecture/social graph obfuscation
	Mantle	FD	–	Passive	–	OOB	–	Pub/ submodel	Group encryption on any storage, pub/sub for interactions
	PrPl	FD	–	Active	Cloud buttler	Undef.	–	Plugins	Cloud buttler either at home or in the cloud/ own language: SocialLite
	Diaspora	FD	–	Active	Hosting nodes	–	–	Hosting nodes	Trusted social hubs, hosting several user pods each
	[Anderson]	D	–	Active	–	PKI	–	Pub/sub	Multi-layer clients with sandbox for external applications
Hybrid	Vis-a-Vis	FD	–	Passive	User pod	–	–	DHT	P2P substrate, data stored in user pods on personal devices/cloud services
	[Kryczka]	D	Social graph, locality	Active	Hosting node	–	Central index	–	Centralized OSN extended with P2P content storage
	[Raji]	D	–	Active	–	BE	–	Pub/sub	Private data on personal storage, rest at OSN provider
	Polaris	FD	–	Passive	User home	–	–	Ext. apps + direct	Storage on phones or servers, NAT traversal necessary
	Confidant	D	Trusted friends	–	–	OOB	Extern. platform	–	Storage on trusted servers, existing OSN is used for signaling (notification)
	Vegas	FD	–	Passive	–	PKI	–	Direct	P2P/reliable storage

the system run on different machines, at different parts of the network and *authorial decentralization* which means distinct and independent authorities run and maintain technical resources.

Since today's OSN have a huge number of users, store a huge amount of data and cause a huge amount of network traffic, they cannot rely on a single machine at a single place in the web. Thus, all OSNs with a certain number of users are technically distributed. Since in Facebook, Google+ and related OSNs one single entity (e.g. company) has the omnipotent power to decide about what is allowed on their OSN platform, we define them to be *centralized* for the rest of the paper.

To be considered as a Distributed Online Social Network (DOSN), a communication system needs to fulfill the two criteria: to provide the core functionality, defined above, and at least one core functionality must not rely on centralized infrastructure. A DOSN is considered *fully decentralized* if this holds for all basic functions.

Abolishing the social network provider along with its infrastructure and data centers raises the question how to organize resource provision. Authors of DOSN approaches offer different answers: decentralization of OSN can be achieved using Peer-to-Peer (P2P) technology or dedicated servers which work together by processing federation protocols. A mixture of both has been suggested as well.

For the rest of the article, we thus define *P2P-OSNs* to be DOSNs that leverage P2P technology to provide the basic functionality, *F-OSNs* to be DOSNs that rely on a client–server infrastructure which is owned and maintained by different authorities and *Hybrid DOSNs* to be DOSNs that rely on a mixture of P2P and client–server infrastructure for different functionalities.

Unreliable resources may cause negative effects on the performance of DOSN. Unavailable storage nodes can potentially be an obstacle for users to access profile data of other users. Thus, the availability of the user data (given as a fraction or percentage of time a data item is available), which is called *profile availability* in the remainder of this article, is an important quality criterion for DOSN. In contrast, the *service availability* means the fraction of time, the social networking service is available to the user.

The issue of profile availability is a particular concern in *P2P systems* where a set of (unreliable) equal nodes are communicating directly without the necessity of server-based message distribution. Participating nodes join and leave the network regularly depending on whether the user uses the service or not. We call this effect *Churn* [50]. It represents the online behavior of nodes in P2P DOSN in the remainder of this article.

2.3. DOSN components

Components of DOSNs are briefly defined below for later use within this article:

User:	**person or organization owning an identifier and a user profile within the OSN.**
User profile:	**digital representation of a user in the OSN, containing all user-owned data items.**
User handle:	**unique identifier for every user being part of the OSN system.**
Content:	**Data item which is stored or shared within the OSN.**
Connection:	**declared affiliation or acquaintance between users (e.g. friendships).**
Node:	**network device, used by users to connect to the OSN.**
Server:	**network device, which reliably supports service provision.**

3. Requirements and adversary models

In this Section, we discuss the requirements that are our benchmark to evaluate DOSNs and introduce the adversary models that are later used to discuss the security of DOSNs.

3.1. Requirements

In the remainder of the paper we present and discuss different DOSN approaches, but none of them has a deployment with a reasonable user base and the full set of functionality compared to today's most popular OSN, Facebook. We consider the non-academic approach, Diaspora, with about 400,000 users[3] to be the most successful DOSN. It still comes without a recommender system for friends and content and without a system-wide content and profile discovery mechanism.

Elaborating success determinants of DOSN is out of the scope of this article, but according to [34], we assume that it is a necessary success-precondition for DOSN to implement attractive functionality in a usable way. Subsequently, we assume that users do not completely trust their OSN providers not to misuse private data [19], but that they do not want to abdicate benefit of OSN functionality. Hence, DOSN need to become as usable and as useful as their centralized counterparts in addition to respect user's privacy to become successful competitors.

In our discussion we thus use Facebook as a baseline for the Quality of Service (QoS). Since we do not have access to implementations of all proposed DOSN systems (the majority of approaches are scientific and thus implementations may even not exist), we discuss the approaches subsequently represented by the set of functionality and performance properties of the proposed DOSN. Furthermore, censorship resistance, security issues and economical issues are discussed in the remainder of this article. We thus assume that DOSN need to provide a comparable level (compared to centralized OSN) of Service Quality in order to be considered as an alternative with respect to:

1. System performance
   - message transfer and profile update delays.
2. Privacy of content and interactions
   - confidentiality and integrity of communication;
   - user authentication and access control;
   - accountability of user actions within the system;
   - incidential data evaluation vulnerabilities (e.g. in case of cipher text access);
   - resistance to censorship;
   - membership concealment.
3. Functionality (e.g. user handle and content search functionality, recommender systems, API).
4. Economical issues
   - network infrastructure costs and storage resource provisioning;
   - type of payment (e.g. money to rent servers or resource contribution via P2P approach).

3.2. Adversary models

The existence of an omnipotent Social Network Provider (SNP) is considered to be a privacy problem by the authors of DOSN approaches. The underlying assumption is that the provider can neither be trusted to protect user data from external attackers nor to withstand misusing the data for monetization purposes. However, the OSN provider maintains a closed system with little attack surface for external attackers. The question thus is whether decentralization is the way to go for improving privacy. To discuss this issue, we define the following set of attackers:

1. An adversary which has read and write access to all data, stored in the system (curious omnipotent SNP).
2. A traffic observer, having an Internet Service Provider (ISP)-like view at the network traffic.
3. An adversary who can enforce all authorities (organizations and companies like SNP and ISP) to corporate with her (governmental attacker).
4. The mass data collector, collecting as much data about as many users as possible (e.g. crawler).
5. The stranger adversary which represents an arbitrary user of the OSN (no direct friendship connection to the attack target).
6. The friend adversary, defined in [25] which is exploiting the friend connection in the OSN.
7. The online reputation attacker, aiming at destroying the reputation of individual users (cyber bullying).

4. DOSN architecture model

The following 3-layer DOSN – architecture model (Fig. 1) introduces an abstraction of the DOSN design space. Subsequently it describes its components which are

[3] https://diasp.eu/stats.html.

addressed by approaches which are covered by this survey or are core functionalities of today's popular OSNs. Existing DOSN approaches individually take just a subset of the optional extensions into account, but minimally specify the DOSN core layer.

The lowest layer represents the communication network which is used for all participating entities to communicate. We assume that it reliably transmits messages from one entity to the other. The middle layer, called ''DOSN – core'' contains all components which are necessary to provide the basic DOSN functionality. The upper layer represents extensions aiming at making the user experience enjoyable. It is divided into two sublayers where the lower part is hidden from the users and thus provides services for the elements of the upper part, facing direct user interactions.

The core component contains three main parts:

- the access control component which can be realized either via access policies, via encryption schemes or a combination of both,
- the profile storage component which describes how profile data is stored in the system and
- an overlay or federation component to organize the communication between nodes. We distinguish between protocols supporting direct user interactions (communication) and those supporting technical information exchange (e.g. profile update propagation) where the latter is transparent to the user and hence may raise different time and volume requirements.

The extensions layer consists of two sublayers. Only the components on the upper layer face direct user interactions while components of the lower sublayer are hidden from the user. We define the following hidden modules of the extensions that implement the extended functionality (Section 2):

- API as an interface for third party applications;
- a recommender system which can potentially recommend both: friends relationships to create as well as content items to consume;
- a search scheme which supports privacy preserving search for user handles or content addresses;
- a social network connector, connecting the new DOSN to existing OSNs, since network effects yield the largest OSN the most attractive since the probability of finding friends is growing with a growing number of users.

The two components with the closest position from user perspective (and hence not hidden from the users) are the GUI (graphical user interface) and the applications which can be built by third parties or users. We consider both to be on the same layer since applications may realize own GUIs.

5. Desing decisions

Representing the core of our DOSN model and understanding the different ideas to address the challenges

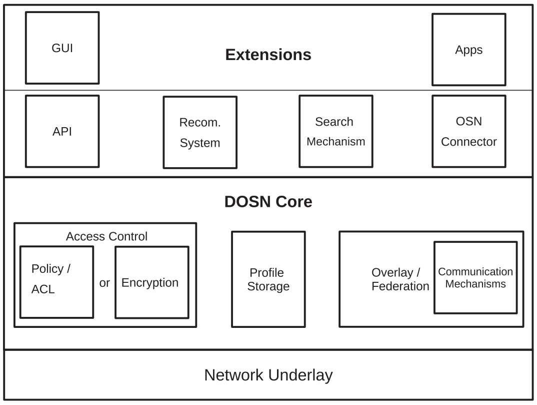

Fig. 1. Architecture for DOSN.

appearing while developing DOSN, lead us to identify the three main criteria for classification: (i) the way of decentralizing the storage of content, (ii) the mechanisms to decentralize access control and (iii) the way how decentralized interaction and signaling mechanisms are implemented.

5.1. Decentralized storage of data

Decentralized storage is a very strong concern of the authors of OSN. The main idea is to put the data owner into a position to hold sway over her PII by keeping the data storage in her influence zone (sphere wherein the user or a trusted third party, e.g. a friend, is directly capable to determine storing, erasure and access operations). In fact, different storage concepts strongly characterize the different DOSN approaches, since they have a big impact on the nature of the architecture itself. Three different fundamental types of decentralized storage of content have been proposed: storing on (i) peer nodes (P2P-OSN), on (ii) external permanent servers (F-OSN) or on (iii) a mix of both (Hybrid OSN).

In the case of *storage on peers* (users' devices), one main challenge is to handle churn provoked resource unreliably. To minimize the risk for data loss and data unavailability, redundant service provision is mandatory. Different performance implications of redundancy procuring approaches yield this design decision crucial for P2P-OSNs. Since replication of resources is the only type of redundancy leveraged in the literature, one important storage-related system design decision for P2P-DOSN approaches is to choose the nodes where to save the copies (i.e. replica placement).

The suggested replica placement strategies are to store:

- data items somewhere randomly in the network;
- data on a set of strangers' devices;
- data on friends' nodes;
- data on a chosen subset of friends;
- data by leveraging a DHT.

To circumvent the availability issue, the replica placement and maintaining effort, storing on reliable *external servers* has been suggested. Profile owners in DOSN have

– in contrast to centralized OSN – the choice where to store their data. The criteria for this decision are monetary cost, trust toward the service provider or guaranteed levels of availability and reliability. We distinguish between flat storage on arbitrary resources and F-OSN based on specialized servers implementing OSN logic at the same place.

Hybrid approaches may allow both: storing data on dedicated servers as well as storing locally on churn-exposed peers.

5.2. Decentralized access control

Since one goal of DOSNs is to improve privacy in OSN (definition in Section 2), DOSN should allow users to define exactly who is part of the set of legitimate content recipients for each single piece of content. Two general primitives have been suggested: access control which is performed by trusted entities as well as encrypting content and distributing keys among legitimate recipients. Furthermore, mixtures of both primitives are part of some approaches.

Approaches relying on ACLs are based on the principle that users have to prove they own necessary rights to an authority enforcing policies defined in the ACLs to access or modify a given piece of content. In our classification, we characterized works relying on ACLs based on who enforces the policies, which can be peers or external servers (based on where data are stored). Finally the ACLs can be enforced by external services in the form of applications or plug-ins, run on the top of the platform, which allow users to interact with each others.

Data encryption approaches are based on the principle that *anybody can retrieve a piece of content, but only users who have decryption keys can interprete it.* Relying on content encryption for performing AC, implies the definition of a key management mechanism. In our classification, we thus characterize several works based on the adopted mechanism.

A motivation for implementing both: an ACL as well as an encryption scheme is that ACL does not protect from access to encrypted content (ciphertext) and thus still allows for inferring communication details like e.g. the data size or communication patterns [26].

5.3. Interaction and signaling mechanisms

Interactions among users in terms of sending messages are at the core of any social platform and may include signaling and notification and establishment of new relationships, etc. In centralized OSNs, the service provider mediates interaction among users.

In DOSNs, interaction mechanisms can be either: (i) still centralized, meaning that they are handled by one single logical entity (in some cases DOSN still rely on classical centralized counterparts for handling interactions [33]) or (ii) decentralized. Decentralized interaction can be realized based on a P2P substrate, relying on direct interactions among user terminals, on the publish-subscribe models or on federation protocols including inter-server communication (e.g. XMPP). Some DOSNs do not define how such mechanisms should be implemented, rather addressing lower level aspects and relying on higher level plug-ins for handling interactions.

In our classification, we thus distinguish between centralized and decentralized handling of interactions and point out the adopted approach.

6. Resulting effects of design decisions

In this section, we discuss the properties and implications of the respected classes entailed by the basic design decision elaborated in Section 5.

6.1. Decentralized storage of data

The answer to the question of where is a convenient place to store data in DOSNs naturally commemorates the impacts of the issues of data availability, storage costs as well as trust. Data availability is an important issue in case of using unreliable resources (P2P). Storage costs become important in case of applying replication schemes that maintain multiple copies in the network or in case of using dedicated resources. In either case, the storage devices have to be trusted to reliably serve legitimate requests and not to leak or misuse accidental data or even the content itself if it is not encrypted.

In P2P-OSNs [10,12,24,5,30,36] data availability is bound to the on-line time of the different principals and can be enhanced thanks to the discussed replication mechanisms. In [11] several replication mechanisms are discussed, which show how availability increases with replication granularity.

No matter how the replication nodes are chosen, storage on peers costs storage as well as bandwidth resources which are not for free. Sophisticated approaches [31,46] aim at minimizing resource consumption while maximizing profile availability. We briefly describe them in Section 9, since these approaches are just replication schemes rather then complete DOSN approaches according to our definition and hence not part of our classification.

However, data replication may affect data consistency, since the latter is significantly harder to achieve as the number of copies of a single piece of content, distributed on several nodes, increases. From the user's point of view, all replication schemes, suggested by the authors of DOSN, come with serious disadvantages:

1. Storing replicas at friends' nodes
   - *Bootstrapping*: it is difficult when entering the network while having no friends.
   - *Correlated failure*: the profile cannot be found by unconnected friends and strangers if all friends are offline at a given point in time.
   - Load balancing is not scalable to popularity peaks if the set of replica nodes is fixed and limited to the friend's devices, assuming that profile data items can be requested publicly (e.g. requests caused by a newspaper article about a person).
2. Random replication without management requires a to high number of replicas to be feasible under realistic churn assumptions [38].

3. Passive replication in which offering access to previously downloaded profiles is granted, does not support unpopular profiles to stay available since unpopular profiles are not frequently accessed.

Assuring data availability and integrity is not an issue in F-OSNs, since users may store a single copy of their data only on a reliable professional storage which they trust for not altering or removing their data. Social network architectures relying on flat storage [21] with no OSN-specific logic implemented, allow *flexible choice of storage resources*, since different types of storage resources (e.g. upload and download services, e-mail boxes or personal web space services) exist. Users may choose external servers for data storage based on criteria such as technical specifications (storage size and bandwidth), trust, monetary costs or reliability. The drawback is that there must be a place to process the OSN logic and if it is not the storage offering server, an additional party, which has to be trusted, is necessary.

In contrast, approaches relying on special OSN servers for storing user data [44,45,43,6] abolish the need for external OSN logic deployment but limit the users in choosing a storage location to the OSN servers instead of any arbitrary storage resource.

Hybrid approaches, allowing both: to store on dedicated servers as well as to store on own hardware (e.g. diaspora), relieve users from the need for external services.

6.2. Decentralized access control

Limiting access to content to a desired set of recipients is at the core of each privacy concept. Three general concepts can be found: ACLs (Access Control Lists), encryption schemes and a combination of both. All those concepts can be realized on the granularity of individuals, role based access control as well as access control on the basis of a group management system.

Restricting access via ACLs can be realized straightforward via granting access to legitimate users after authentication (before accessing content, the knowledge of the secret needs to be proven) or identities (users being part of a content owner-defined set of legitimate identities are allowed to access).

Since ACLs do not provide any kind of protection against attackers which are able to listen to the communication at the underlaying network and access policy enforcing parties need to be trusted, several authors of DOSN suggest to implement encryption schemes in spite of their need for key distribution. In our classification, we thus characterize the approaches based on the trusted parties enforcing the ACLs and the adopted key management mechanism in case of content encryption.

Most encryption based DOSNs [54,21,7,6] rely on Out-Of-Band (OOB) mechanisms for exchanging the whole keys (or fingerprints of the key that can be used for retrieving the associated key, for example relying on cryptoIDs [39], as suggested in [6]). This of course implies the disadvantage of the need for a secure and trustworthy OOB channel.

As an alternative to OOB mechanisms, Safebook [13], PeerSoN [10,51] rely on trusted nodes playing the role of *Credential Authorities/PKI*. In those cases, the Credential Authorities (CA) are only used to cryptographically initiate the OSN, while they do not mediate communications and cannot trace interactions.

Finally, Graffi et al. in [24] propose to rely on a DHT substrate also for distributing keys, which allows to avoid any necessity for a central node in the network. As a consequence, there is no need for OOB communication nor Credential Authority, but it comes without any kind of identification.

Cryptographic methods which allow malicious parties to access cipher code still do not prevent from inference attacks. Attackers may infer the type of a message (e.g. video vs. chat) from the size. Furthermore, access to cipher code allows attackers to notice actions (depending on the encryption mechanism) like revocation of access rights or access patterns [26]. A combination of limiting access via ACLs and encryption or obfuscating methods like chunking and salting can mitigate this issue.

6.3. Interaction and signaling mechanisms

Different types of interaction handling mechanisms with contrary implications have been proposed by the authors of the approaches which are covered by this article. Interaction includes messaging as well as sharing pieces of content. Sharing operations consist of making pieces of content available for being downloaded by other users. We distinguish between centralized and decentralized interaction mechanisms.

6.3.1. Centralized handling of signaling

Centralized interaction handling can be achieved by purpose-built specialized services or via utilizing existing OSNs (e.g. Facebook) for signaling.

The centralized interaction mediation and handling of metadata is suggested in [32]. In contrast to centralized OSNs where resources of a central authority takes care of different functionalities (e.g. storage, authentication and interaction management), a (single) central node is responsible just for the interaction and metadata handling. The aim is that interaction handling can be done via reliable and powerful units for achieving good quality of service without having a central authority which is able to access user data. The underlaying assumption is that user data access (like in centralized OSNs) is a crucial part of the service provider's omnipotence and needs to be abolished. Lockr [53], Polaris [54] and Confidant [33] rely on *existing platforms* performing signaling mechanism.

However, centralized interaction and metadata handling approaches still give the OSN provider access to metadata and content access information. That means the central authority is still able to learn e.g. the interests, social connections and popularity of users and their profiles. Hence these approaches require the users to trust the authority to a certain level.

6.3.2. Decentralized handling of interactions

P2P systems often rely on a DHT as a signaling mechanism thus mediating interactions. For example, in Safebook [13], PeerSoN [10] and Vis-a-Vis [47] the DHT system can be used as an asynchronous messaging mechanism, which may include signaling of new actions/interactions. Users can query the DHT with the ID of a user or a specific piece of content.

A unique feature of P2P-OSNs consists of the possibility of direct interactions among users, also with no Internet access, how suggested in PeerSoN [10] and in Polaris [54], at least for some kind of interactions. While PeerSoN relies on direct data exchange among user devices, in Polaris data may be stored/replicated on external servers and user smart phones only play the role of entry point to user data.

The decentralized interaction handling comes with the main advantages not to require trust into a single entity for that (interaction) purpose. Drawbacks are that those decentralized systems may leak metadata by cipher evaluation [26]. They may also suffer from churn-caused node unavailabilities and – like approaches with purpose-built centralized interaction mechanisms – from missing connectivity to popular centralized OSNs like Facebook.

7. DOSN approaches

This section supplements the Table 1 with a short paragraph of text for each approach. The rationale behind this section is that a classification cannot capture all unique details of all approaches. The aspects which are already covered in the classification are not mentioned again, except when they are part of the unique clue of the approach. Advantages and disadvantages of approaches are not discussed in this section for each approach, since similarity of approaches causes redundancy in the evaluation. Section 8 encloses an evaluation based on classes instead (Fig. 2).

Furthermore, we explain the publication time line aiming at making it easy to grasp when an approach was published and which approaches can be assumed as known by which authors of newer approaches.

7.1. P2P-OSN

7.1.1. PeerSoN

The authors of PeerSon [10] propose a two-tier architecture in which the first tier is a Distributed Hash Table (DHT) and the second tier consists of the nodes representing users. The idea is to use the DHT to find the necessary information for users connecting directly to the target nodes. This approach comes without a replication scheme and stores offline messages at the DHT (OpenDHT in the prototype implementation). All user content is encrypted.

7.1.2. Safebook

Main objective in case of Safebook [13] is to protect privacy of users in a DOSN setting. The architecture consists of three main components, namely: Matryoshkas (a ring-like ego graph reflecting friendship relations), a P2P lookup service and a Trusted Identity Service (TIS).

Each node is surrounded by its friends (first shell) and friends-of-friends (second) shell in its Matroyschka. User profiles are replicated for better profile availability at friend's nodes in the innermost shell. Nodes at the outermost shell are entry points for routing requests to the center of the Matryoshka and can be found via querying the lookup service. This overlay structure hides the friendship relations from strangers by multihop routing. TIS verifies user identities.

7.1.3. LotusNet

LotusNet [5] is a modular P2P-OSN platform, realizing social network functionality in widgets. The communication infrastructure as well as the encryption scheme and the identify management is realized by using the DHT modification Likir [4]. Access control is realized by signed grants for proofing social relations. The data owner hence specifies the type of social relation which is necessary to access the data item.

7.1.4. LifeSocial.KOM

Graffi et al. [24] present an approach where all OSN functionality is realized by plug-ins. Storing and exchanging data items is realized with the help of FreePastry [41]. PAST [17] is used for data replication. Cryptographic Public Keys are leveraged to be user IDs in the network thus uniquely identify users in addition to encrypt content and messages.

7.1.5. DECENT

DECENT [30] is a modular and object-oriented architecture DOSN architecture. It leverages a DHT to store user data and uses cryptography to protect confidentiality and integrity of user-owned content. The focus of the authors is a blog-like wall rather than chat messages. The architecture is modular, i.e. the data objects, cryptography and DHT are three separate components interacting with each others based on an interface. This modularity causes freedom to use any kind of cryptography (ABE based on [29] is suggested in DECENT) and any type of DHT.

7.1.6. Cachet

Cachet [36] is an improvement of DECENT. Thus it is also a decentralized architecture for social networks that provides strong security and privacy. The main difference is that Cachet introduces social caches to improve the performance of the system by avoiding the pull-based grasping of many single data items from different sources. Therefore nodes leverage social trust relationships to "maintain continuous secure (SSL) connections with online contacts to receive updates directly as soon as they are produced". In case of overlapping online times, this type of presence protocol can effectively reduce communication delays.

7.2. F-OSN

7.2.1. SoNet

SoNet [44] circumvents the implications of P2P mechanisms (like profile availability and free-riding attempts in resource provision) by suggesting an XMPP-like architecture. Every node is attached to one server, implying the address scheme to be user@host (RFC 822). Profile data is

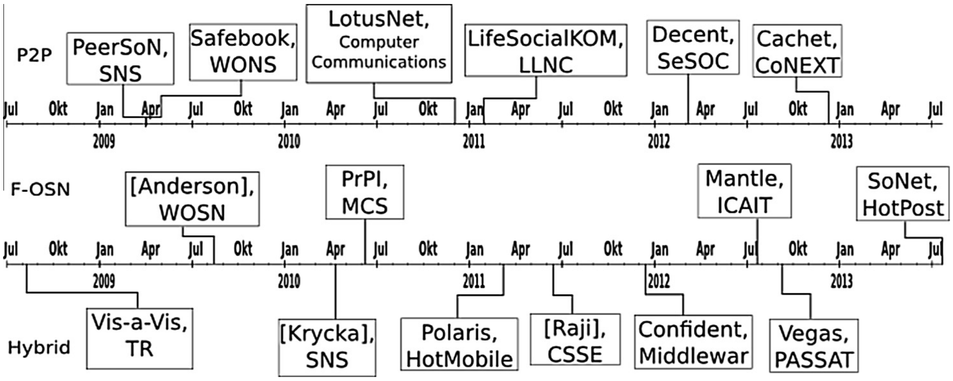

Fig. 2. Time line of the publication dates.

encrypted and replication is still part of the architecture to mitigate server failures. The clue of this approach is to obfuscate the social graph by introducing single direction pseudonyms.

7.2.2. Mantle

Mantle [21] is a DOSN approach, settled around the idea to leverage arbitrary storage in the web (cloud services as well as mailboxes, etc.), to store user data. Since the arbitrary storage concept disallows storage entities to deploy any logic, the service-related logic is implemented in user-owned clients. Interaction is managed by employing a publish/subscribe model and is handled locally without any help of centralized server.

7.2.3. PrPl

PrPl [45] stands for Private Public. The main goals are to allow users to store data in their own influence zone by choosing trusted storage resources and to run social applications across different domains while sharing data without privacy concern. The idea of the architecture is to have Personal Cloud Butlers to store personal digital assets to support access control mechanisms. A Pocket Butler handles all authentication and communication with personal cloud Butler along with the facility to allow sharing of resources across multiple applications.

PrPl uses SociaLite: a language based on Datalog which allows developers to access the data by just querying on the data served by Butlers. OpenID is used for authentication.

7.2.4. Diaspora

The main aim of Diaspora [43] is to build a reliable and usable decentralized online social network. The architecture is based (similar to PrPl) on a client–server model where every user has his own server instance (Pod) which is used for storage, communication and access control. Since there is no data or service replication, pods must always be online for reliable service provision. A Pod can be hosted either on own hardware or by a service provider (cloud service) Data is stored unencrypted on the pod, protected by an access control mechanism.

7.2.5. [Anderson]

The authors of [6] define a privacy preserving architecture for decentralized social networking that takes advantage of the simplicity and performance of the centralized client–server model. The main goals are to protect personal data from unauthorized access, to hide the social graph (like friendship links) as well as assuring content integrity.

The ideas described in this approach are closely related to the field of software engineering rather than network architecture. The authors suggest the client software to consist of the following layers: the application layer, the data structures layer, the cryptographic layer and the network layer. The layered architecture render the software components on each layer exchangeable. All applications are supposed to run in a sandbox, allowing the applications to access just a predefined subset of the private data.

7.3. Hybrid DOSN

7.3.1. Vis-à-Vis

A VIS (virtual individual server) [47] is a reliable personal server, assigned to every user to store her data. The main idea is to build overlay networks (with VISs as members) that correspond to social groups. Members of groups are supposed to have the intention to share their location. The focus of Vis-à-Vis is to support location-based OSNs while preserving privacy of location information by supporting flexible degrees of location sharing in different groups.

7.3.2. [Kryczka]

In this approach, a User Assisted OSN (uaOSN) [32] is proposed where users can contribute resources to reduce the costs of the OSN provider and to increase scalability. In uaOSN, queries are sent to the provider which informs the user about storage placement for large content items like photos or videos. The uaOSN provider stores the user profiles and metadata of the outsourced content. Data in uaOSN can be either stored on user's desktop or set top box/residential router which can have a hard disk or on paid storages like Amazon cloud services. To achieve better a profile availability, data is replicated. An encryption scheme is not part of this approach.

7.3.3. [Raji]

Similar to the uaOSN, Raji et al. propose in [40] to store private data (encrypted) beside the OSN on personal storage servers which are assumed to be honest but curious. A BE scheme enforces the access control as well as the confidentiality of the data.

7.3.4. Polaris

Polaris [54] is an "architecture for OSNs that preserves monetary incentives for OSN providers to store and manage user data, while also mitigating the systemic privacy concerns associated with monolithic OSNs." To realize this, a user can choose a different provider for each functionality (e.g. photo storing or microblogging). Highly sensitive data is stored at a mobile phone which is assumed to be able for keeping small pieces of content available. The authors argue that as a result every provider which is involved in service provisioning can just access a subset of the whole personal data.

7.3.5. Confidant

Confidant [33] fosters decentralized data-processing being scalable and affordable by storing data without encryption. It relies on social trust relationships in friends to replicate the data on secure devices. The challenges addressed in this paper are to manage access control and to assure data consistency among the distinct replicas.

7.3.6. Vegas

Vegas [18] is a DOSN architecture proposing to use reliable data storages for increasing the availability of user data in a P2P setting. The encryption scheme based mutual public keys for exchanging symmetric keys is used to ensure the confidentiality of user data.

8. Evaluative discussion

In this Section, we discuss the present situation in the field of DOSNs on the way to become an alternative to their centralized counterparts. We thus elaborate the degree of achievement with respect to our requirements. We focus on the fitness of the DOSN approaches to help to improve privacy as well as on the quality of user experience. We discuss the latter by looking at performance issues, resource provision, technical knowledge which is necessary to use the DOSN and finally the offered functionality.

8.1. Privacy and security

The major reason for authors to suggest a distributed approach for social networking is to increase privacy and security. Hence, the enthralling question is: Are the suggested approaches appropriate to achieve better privacy and security? We discuss this question with respect to our adversary models (Section 3.2).

8.1.1. SNP attacker

The overwhelming majority of approaches abolishes the SNP completely and hence it does not exist as attacker anymore.

The user-assisted OSN (uaOSN)-approach of Kryczka et al. uses user devices for data storage. Even though the users may be able to exactly specify which data is sensitive (we doubt that, since sensitivity is depending on the knowledge of the attacker) and store this data on private storages, the OSN provider is still able to learn sensitive facts about the users by evaluating incidental data. The SNP may learn habits like diurnal usage patterns (e.g. conclude that the user works at night) and the social graph. That issue applies for Polaris in the same way, since every provider of a particular functionality can learn usage patterns and two-sided actions like messaging (sender and receiver) potentially leak knowledge about social relationships. We argue that it is necessary to hide metadata and communication habits as well as social graph information from the SNP to protect user's privacy.

8.1.2. Traffic observer

The situation in the field of P2P-OSNs is that all approaches implement some kind of encryption. Assuming that the attacker is not able to decrypt ciphers, she is still able to infer who communicated with whom and how often. Furthermore, the data item sizes can be inferred by observing the traffic. This allows for guessing what kind of data is exchanged (chat messages, photos or videos). If replication schemes rely on social graph metrics (e.g. friendships or trust), those can potentially be observed as well. Only one approach, covered by this survey (Safebook), tackles these issues by redirection schemes or traffic obfuscation.

Safebook introduces the concept of Matryoshkas where friends form ring-like structures in egocentric networks. Traffic is redirected from outer to inner circles. Nevertheless, Matryoshkas are still vulnerable to timing and traffic observation attacks since there is no traffic obfuscation or message throttling included.

Inferring facts by observing traffic in F-OSN can be challenging if more than one user is using one server and if the servers are re-encoding the data items, since the traffic observing attacker can than only observe that a set of users is connected to the server but not who exactly communicates with whom. The success of the attacker depends on how much information can be learned from communication intensity (traffic, data size) and timing attacks.

Considering hybrid solutions, the situation strongly depends on the concrete architecture. They are as vulnerable to traffic observing attacks as P2P solutions are if direct communication happens among peers or if it can be inferred from the storage place to which the latter is assigned. Approaches like Polaris [54] may mitigate the success of traffic observers by combining different centralized services for communication and storage. The uaOSN [32] cannot be evaluated by now, since it is highly dependent on what exactly is stored at the peers and how it is accessed. A caching mechanism in the centralized part of the uaOSN can be a game changer for traffic observing.

8.1.3. Governmental attacker

Governments tend to try to censor or even disable social networks as soon as riots start in that country. The Arab spring is a prominent example for that phenomena. Governmental type of attacker can (simplified) be considered being a unification of the SNP attacker as well as the traffic observer. The questions is: Is there an approach which can resist the governmental attacker?

Even though assuming that the government does not want to disable the whole communication infrastructure of a country to disable social networking, we argue that none of the approaches is bullet proof. P2P approaches (without traffic obfuscation) are vulnerable to traffic observing attacks: Governments could find out who communicates with whom and who is important for organizing demonstrations. Server-based architectures can easily be deactivated if the servers are well known and run within the influence zone of the government. From our point of view, research in making DOSN more resilient against governments is eligible.

8.1.4. Stranger

Stranger attacks with the goal to learn private facts about a particular user would be very weak if people would use the access control mechanisms in OSNs properly and if they would be aware of inference attacks (assuming facts to be valid also for friends). The success of stranger attacks are in general less depending on the architecture of DOSN but rather on efficient access control. Perfect usage assumed, none of the presented DOSN open an attack surface for stranger attacks.

8.1.5. Mass data collector

State of the art for mass data collection is building crawlers. That could be possible if data items are publicly-accessible and if user handles are available to address profiles. The straightforward approach for crawling a social network is to first create a user account then initially connect to arbitrary users and try to crawl their friend lists.

Iterating over friend lists can theoretically lead to a discovery of the whole connected region if every user allows to access a list of her friends. Hence, no matter at which type of architecture, it is very important to disallow strangers accessing friend list in general. DOSN architectures thus do not help to mitigate attacks from mass data collectors in case that access to profiles is restricted in centralized OSNs.

8.1.6. Friend attackers

Friends, being attackers aiming at accessing more information than authorized by the data owner, can be successful either when access control is not performed properly or if replication schemes in P2P-OSN rely on social trust. A node where friend's (encrypted) content is stored can still learn incidental data.

8.1.7. Cyber bullying

The social phenomena to attack the reputation of an individual can be observed in an environment like OSN as well. In a centralized setting without content encryption, the omni-potent provider can delete content as well as user accounts if the well-behavior rules are not respected by users. No author of a DOSN considered misbehavior of users by now. To tackle that issue, accountability of actions (e.g. posting content or messages) needs to become a focus of DOSN. Accountability, however, may affect the achievement of anonymity goals.

8.1.8. Conclusion for the privacy and security evaluation

The main advantage in privacy protection, which can be achieved with the approaches in this survey, can be seen in protecting against the central OSN provider. Authors tend to protect communication content rather than hiding communication. From our point of view, no approach can protect against the governmental attacker being interested in building communication graphs.

Censorship is not an issue in most DOSN (in Polaris and uaOSNs it still is) because of the usage of cryptography. That implies that it is hard to frustrate illegal actions. Illegal content can be shared with only a minimal risk and attacks on the reputation of users are abetted. The answer to the general question whether DOSNs improve privacy and security of users strongly depends on the point of view: DOSN can effectively protect the content which is shared but they may foster misusage.

8.2. User experience

Measuring user experience by conducting a user study in DOSN is hard to perform since no DOSN (except Diaspora) has a user basis rather than public-available and usable prototypes. Having no better alternative, we discuss performance issues, the skills being necessary for using DOSN and the functionality instead.

8.2.1. Performance

P2P approaches replace the database queries in centralized OSN by vast inter-node messaging for accessing data. The authors of Safebook [13] consider 11 s to be a realistic time for requests if no performance optimizations are employed. Cachet introduces a caching strategy for improving the performance by maintaining encrypted channels to friends. Receiving unpredictable data items from strangers (e.g. friend-of-friend) still causes time consuming operations.

F-OSN and Hybrid approaches do not suffer from these P2P-specific performance limitations, but still cause federation overhead. In general we would see a performance advantage for centralized OSNs since a single authority is able to globally optimize its databases and to build caches.

8.2.2. Usability

If DOSNs leverage cryptography for privacy, basic knowledge about cryptography may be necessary to use the DOSN. For example, users need to understand that they need to exchange public keys. Furthermore, the presented approaches (except Polaris) require users to install a client software instead of being only a web application. Installing software on local machines may cause a need for actions which require administrative rights on the local system. Moreover, the necessity of local software installations could cause interoperability issues and is an additional procedure which might be an obstacle for users. We thus argue that any kind of DOSN should be running at every web-connected device without installation obstacles to achieve a usability which is comparable with centralized OSNs.

8.2.3. Functionality

No DOSN provides the same functionality like Facebook. One reason of course is that the approaches are academic and concentrate on a particular ideas to present rather being intended to be a social network which can be used by the users.

The second reason is more wholesale in nature: popular functionality like recommender systems, search functionality and some online games leverage the social graph and user attributes. Having only local knowledge, the whole social graph is not known to any single entity. The local (egocentric) structure can be learned via exchanging messages, functionality based on global knowledge will be too expensive.

Acquiring the graph knowledge beyond the own graph neighbors (friends) via messaging may effect the user privacy. Even paramount functions like a sophisticated search mechanisms for user handles is not available in a privacy preserving manner.

8.2.3.1. Conclusion for the user experience evaluation. Considering our metrics for the user experience, the presented approaches (except uaOSN) will suffer disadvantages in comparison with centralized OSNs. This holds for performance as well as for the skills and the functionality. We consider F-OSN and Hybrid DOSN to have the biggest potential to present a good trade-off: they can be web-based as OSN are nowadays, do not suffer the performance limitations caused by maintaining P2P structures, and in case that the majority of friends is assigned to one server, they allow efficient local operations instead of grasping data items from a high number of different sources.

9. Related approaches

We discuss some approaches in this Section which address DOSN related issues or approaches only aiming at improving specific sub-aspects of DOSNs to highlight current challenges. We include approaches addressing the profile availability issue in P2P DOSNs, encryption schemes for DOSN as well as for centralized OSN being one alternative to distributing OSN and finally we include social network integrators.

Social network integrators are also included, since they offer potential ways of extending the initially limited user basis of DOSN to temporarily increase their attractiveness until sufficient adoption. We thus think that a social network connector is a crucial success component for new upcoming social networks.

9.1. Profile availability in P2P OSN

The load and requirements for storage in P2P-OSNs differs strongly from distributed storage as well as from file sharing. OSNs environments require the storage layer to reliably store many unpopular content items which are frequently updated. This is in stark contrast to file sharing applications, which usually provide a comparatively low number of large and popular files to a high number of users. A stark contrast exists even to conventional P2P backup and storage scenarios, which are characterized by rather infrequent I/O access to the stored data. Furthermore, contrary to file sharing and P2P backup and storage, in which all participants are treated somewhat identical, OSNs contain information about friendship and trust relationships that can be exploited. Many techniques that are deployed in P2P storage environments – like erasure codes – are not convenient in this dynamic environment. Thus, none of the P2P-OSN approaches is based on a file sharing nor P2P storage scheme. The following subsection explains the solutions for profile availability in P2P-OSNs that are discussed in the literature.

As shown in [38], choosing replication nodes randomly leads to a high number of replicas if a convenient availability of user profile data should be achieved. Friend storage approaches suffer from localization effects: if all friends live in the same time zone (e.g. same city), it is very likely that they have the same off-line times in the night. Furthermore, if a new node has no connections to friends, it does not benefit from replication. Choosing the best subset of friends for profile replication is an NP-hard problem [49].

Finding a systematic solution for having a good availability with minimum of cost and overhead is the goal of the authors of MY3 [35], SuperNova [48], Gemstone [52], SOUP [31] and S-Data [46].

SuperNova introduces super nodes for bootstrapping and circumventing the disadvantages of utilizing friends' nodes for storing. Gemstone has a metric-based approach to select some friends which are a good choice for achieving a high availability with low costs and SOUP proposes to select replica nodes by calculating an online experience among friends. S-Data is a group-based approach where groups are generated on the basis of diurnal online patterns to reduce the number of replicas. The authors of MY3 [35] propose users to choose a subset of friends (trusted proxy set) for performing profile replication and access control. Arguing that trusting the friends which are performing the access control can replace the encryption of content and hence abolishing key distribution. This assumption simplifies the approach.

9.2. Encryption schemes for OSNs

Abolishing the omnipotent and trusted social network provider as a mediator of communication between social network users, combined with the introduction of replication schemes in P2P-OSN, leads to the need for encryption of content and communication (in case of leveraging untrusted resources). F-OSNs or hybrid solutions often aim at mitigating the need for trusting server entities and rely on cryptography for this purpose. Only a minority of DOSN approaches comes without encryption and relies on trust in friends (e.g. My3 [35]).

Thus, the efficiency, performance and usability of encryption and key distribution schemes are crucial factors for DOSNs for being widely adopted. For this reason, some authors work on building new cryptographic mechanisms for that specific issue. Mentioning their relevance, we present a brief sketch of the ideas and a brief overview of this field.

9.2.1. Brief OSN encryption background

Mutual public keys are the straight forward type of realizing an encryption. For every recipient of a message, one encryption procedure has to be performed. Asymmetric cryptography, however, is comparatively expensive compared with symmetric cryptographic algorithms. Group key management mechanisms based on symmetric keys tackle that issue by distributing one symmetric key among a group of recipients (e.g. via mutual public key schemes). Hence, one (symmetric) encryption process is sufficient to share content with a group of recipients. As long as the group setting does not change, a new key distribution is not necessary.

Broadcast encryption (BE) schemes can rely either on symmetric or asymmetric encryption and are used by senders to share confidential data with a dynamic set of recipients in a cost-effective way. BE requires each recipient to have an individual key. In BE schemes, the encryptor uses an encryption mechanism that allows to produce ciphertext that can be decrypted by plenty of keys which are defined during the encryption process. If a private key generator, leveraging identities to decide about legitimation is part the system, the broadcast encryption scheme is called identity based broadcast encryption (IBBE) [15,9].

Attribute based encryption (ABE) [42,23] schemes adopt that idea. An encryptor decides who is able to decrypt the ciphertext by labeling the latter with a set of descriptive attributes. Private keys are associated with ACL structures to decide, based on those attributes, which ciphertexts can be decrypted. The encryptor thus does not decide about decryption by taking single keys or identities into account but defines attributes or combinations of attributes which a decryptor needs to meet to be able to decrypt a message.

9.2.2. (D)OSN encryption contributions

The main goal of Persona [7] is to disallow third parties to access personal information by deploying attribute-based encryption (ABE) in an OSN context. Each user generates an ABE public key (APK) and an ABE master secret key (AMSK). For each friend, the user can generate an ABE secret key (ASK) corresponding to the set of attributes that defines the groups that friend should be part of.

The main contribution of the model from Sun et al. [51] compared with Persona is to have a very efficient revocation of content access rights. It uses broadcast encryption that enables the data owner to exercise desired access control.

The authors of Noyb [27] (“none of your business“) suggest to improve privacy by encrypting content and to modify it in a way that it looks like legitimate content. Hence, it allows users to use existing OSN while disallowing provider to access the content. Applying this approach is not an obstacle for the provider to learn usage patterns as well as friendship relationships.

Günther et al. [28] provide a building block for privacy preserving treatment (including encryption) of user profiles in OSNs. The authors of [8] compared different encryption mechanisms and evaluated them for their applicability in the DOSN context and concluded that broadcast encryption would be the best choice for this use-case.

EASIER [29] is an ABE architecture for DOSN, supporting dynamic group memberships and revocation of rights without re-encrypting data items or issuing new keys. The main idea is to introduce a proxy which needs to be contacted before decryption. A user sends a part of the cipher text (CT) to the proxy where a transformation takes place. The transformed CT can only be decrypted if the right was not revoked.

Lockr [53] is an identity-management tool for OSNs that allows users to codify their relationships through social attestations. The primary goal is to provide privacy as well as to simplify site management and accelerating content delivery. Lockr's decoupling eliminates the burden on users of maintaining, several up-to-date copies of social networks, performing user-ID reconciliation across sites, and familiarizing themselves with the varied access control mechanisms provided by each site. A social attestation is a piece of data that certifies a social relationship. By issuing an attestation, the issuer tells a recipient that they have formed a relationship.

The presented encryption approaches show that retaining confidentiality of content in OSNs is possible. In case of suggesting a new DOSN approach, authors may rely on existing encryption mechanisms.

9.3. Private discovery of common social contacts

Discovering common social contacts is a common feature in today's OSNs like Facebook. In privacy preserving distributed systems, it may not be desired to exchange contact lists. De Cristofaro et al. [14] introduces a scheme which allows for finding common friends without disclosing non-common friends. To the best of our knowledge, there is no discovery mechanism which neither disclose the search index nor the search queries. As a result, there is a trade-off between implementing (or using) a discovery mechanism or preserving privacy with respect to search index and queries.

9.4. Social network integrators

OneSocialweb[4] is a project aiming at building an XMPP-based connector which potentially integrates all OSNs into one large social network. Other social network integrators, connecting a subset of popular services are:

1. Meople (http://meople.net/, accessed on 20th of January 2014) aggregated SNSs: Facebook, LinkedIn, Google+, Twitter, Instagram, YouTube, Flickr, Groupon, Tumblr, Foursquare, VK, Odnoklassniki.

[4] http://onesocialweb.org/, accessed on 20th of January 2014.

2. Jyst (http://jyst.us/, accessed on 20th of January 2014) aggregated SNSs: Facebook, Twitter.
3. Alternion (http://www.alternion.com/, accessed on 20th of January 2014) aggregated SNSs: Facebook, LinkedIn, Twitter, Instagram, YouTube, Flickr, Foursquare, Picasa and the mail accounts: Gmail, Hotmail, Yahoo!, AOL.
4. Yoono (http://www.yoono.com/, accessed on 20th of January 2014) aggregated SNSs: Facebook, LinkedIn, Twitter, YouTube, Flickr, Foursquare, MySpace, Yahoo!, Google Talk, AIM, FriendFeed.
5. TweetDeck (http://www.tweetdeck.com/, accessed on 20th of January 2014) aggregated SNSs: Twitter, Facebook, Myspace, LinkedIn, FourSquare, GoogleBuzz.
6. Hootsuite (http://hootsuite.com/, accessed on 20th of January 2014) aggregated SNSs: Facebook, LinkedIn, Foursquare, MySpace, PingFm, Wordpress.
7. SpredFast (http://spredfast.com/, accessed on 20th of January 2014) aggregated SNSs: Facebook, Twitter, LinkedIn, Google+, YouTube.

The aforementioned social network connectors could potentially been leveraged to bootstrap a new (D)OSN since the attractiveness of OSNs is strongly bound to the user basis.

10. Impacts on stakeholders caused by the decentralization of OSN

In this Section, we discuss the impact of decentralizing OSN on today's OSN stakeholders. This includes benefits, drawbacks and challenges.

The stakeholders in the context of OSN, considered in this work, are: the OSN users, the OSN provider, the advertising companies which benefit from utilizing the advertising opportunities offered by the OSN provider, the governmental state and media consumers which are not necessarily part of an OSN. Effects on extenders (e.g. application sellers) like Zynga[5] are not discussed since the effects on them is strongly depending on the particular architecture.

Since we consider the user to be the most important affiliate, we start our discussion with the following benefits of OSN decentralization for the users:

- *Ownership:* Facebook and other OSN provider ask the users to transfer the copyrights of any content from the user to the OSN owner. In contrast, decentralization holds user data in the influence zone of the users (Section 2). The copyright transfer can be avoided.
- *Privacy:* In centralized OSNs, users need to trust the omnipotent provider not to misuse the data and to be able to protect the data from attackers.
- *Flexible choice of resources:* Building, running and maintaining OSN platforms cause expenditures. In centralized OSNs, platform-related resources are provided by the service provider itself. In most cases, they present no monetary costs to the final users [20], which pay by agreeing for such platforms to exploit their data with a commercial purpose.

  One of the benefits a decentralized approach should bring to users, is that they should have the possibility to choose what resources to rely on. For example, a user can choose whether adding the own device's resources (e.g. P2P approaches) or relying on dedicated servers (e.g. Diaspora). Using dedicated resources for building DOSN does not cause an exploitable dependency like using the resources of centralized OSN provider, since the OSN membership does not require an affiliation with one specific authority. Hence, the resource provider is replaceable.

  This opens to several business models which strongly differ from exploiting user data for commercial purposes.
- *No censorship:* Several centralized OSNs perform active censorship[6,7] – called decency or content control – on user behavior and content. This imposes significant limitations on what a user can or cannot do within the platform, based on rules which may strongly vary from one platform to another and are in general very subjective. Such rules can also be very country-specific, since OSNs have already accepted to be compliant to local laws imposed by several governments.
- *Openness:* DOSNs abolish the central authority, causing that no single authority is able to exclude a user from the platform by suspending his account (e.g. because of not accepting copyright transfer rules).

While decentralizing OSNs does not come without drawbacks, the following aspects may become an issue for users:

- *Resource provision:* Centralized as well as decentralized OSNs need technical resources (e.g. storage, bandwidth) to operate. In centralized OSNs, the provider is responsible for making them available. The most popular approach to compensate these efforts is to sell opportunities for personalized advertising. In the decentralized case, other mechanisms need to tackle the resource issue.
- *Profile availability:* Availability of user profiles is strongly linked to the architecture of an OSN. In the centralized case, the provider takes care of storing and keeping user profiles available. Federated DOSNs rely on independent professional and reliable resources while P2P OSNs utilize the unreliable resources of the users devices.
- *Cyber bullying:* Since decentralized OSNs hide or encrypt the communication, no central authority can enforce rules. In the real world, no administrator can switch off the voice of people which are bullying others as well,

[5] http://zynga.com/, accessed on 06th of July 2014.

[6] http://www.facebookcensorship.com/, accessed on 20th of January 2014.

[7] http://www.dailytech.com/Google+Plays+Name+Police+Conducts+Baffling+Censorship+Crusade/article22238.htm, accessed on 20th of January 2014.

but everybody is responsible for what he or she is doing. We argue that DOSN should support accountability to protect users.

- *Metadata privacy and the concealment of communication partners:* Depending on the particular architecture, decentralizing OSNs may raise security issues that do not exist in centralized OSNs [26]. An attacker which is able to observe traffic in the underlying communication network (e.g. IP) could track who communicates with whom if OSN devices are assigned to a single user and can send messages to other devices without obfuscation. In contrast, a centralized OSN has a mixing functionality. Assumin frequent usage, mixing functionality implies that the observer could only find out that users communicate with the provider but not track single communication paths among users.
- *Functionality:* OSNs allow to build social applications which are based on interactions between users who share social links or special interests. Examples for these applications are recommender systems, interest matching algorithms for mediating between users as well as games. Due to the nature of decentralized systems, the complete social graph as well as the complete set of interests of all users is not known to anybody. Each node has only local knowledge. Hence, the graph knowledge needs to be grasped using federation protocols, if it is necessary for an application.

Other affiliates become more affected by economical issues. Abolishing the central OSN provider naturally destroys their business model and hence other companies cannot benefit any more from utilizing the advertising opportunities.

11. Summary and conclusion

In this article, we discussed the impact of decentralizing OSNs on different OSN affiliates of today's popular OSNs, explained the design space by introducing an architecture model, presented and discussed a classification of DOSN approaches and introduced some DOSN-related approaches. Finally, we presented the unique ideas of each discussed approach.

Decentralization of OSN can tackle two important issues: First, it is a possibility to circumvent the need to trust the SNP for not learning facts which cannot been hided by encryption. An omnipotent provider could still learn who communicated with whom and how often. Second, users do not need to accept copyright transfers to the SNP and terms of usage which are disadvantageous for them.

The result from security perspective is that DOSNs mainly aim at protecting content from curious provider and assuring confidentiality of user communication. DOSNs potentially abolish content censorship by leveraging encryption schemes Beside Safebook, none of the presented approaches introduces mechanisms to protect against traffic observer or governmental attackers building communication graphs.

The result of our functionality discussion is that the discussed ideas do not solve the issue of providing attractive social-graph based functionality like a comprehensive privacy preserving search and a recommender system like in the centralized OSN counterparts. Hence, we argue that the field of DOSN could benefit from research in building privacy preserving graph-based functionality, combined with performance optimizations.

Acknowledgements

This work has been [co-]funded by the German Research Foundation (DFG) in the Collaborative Research Center (SFB) 1053 ''MAKI – Multi-Mechanism-Adaptation for the Future Internet''.

This research in parts has been funded by the MSIP (Ministry of Science, ICT & Future Planning), Korea, in the ICT R&D Program 2013.

References

[1] Censorship in Facebook. <http://www.rechtzweinull.de/archives/840-Zensur-bei-Facebook-Reichweitedes-virtuellen-Hausrechts.html> (accessed 2014-01-16).

[2] Censorship of Femen in Facebook. <http://www.mmnews.de/index.php/etc/13534-facebook-zensiert-femen> (accessed 2014-01-16).

[3] Facebook Censorship in Syria. <http://www.reuters.com/article/2007/11/23/us-syria-facebook-idUSOWE37285020071123> (accessed 2014-01-16).

[4] Luca Maria Aiello, Marco Milanesio, Giancarlo Ruffo, Rossano Schifanella, Tempering Kademlia with a robust identity based system, in: IEEE P2P, 2008.

[5] Luca Maria Aiello, Giancarlo Ruffo, Lotusnet: tunable privacy for distributed online social network services, Comput. Commun. 35 (1) (2012) 75–88.

[6] J. Anderson, J. Bonneau, C. Diaz, F. Stajano, Privacy-enabling social networking over untrusted networks, WOSN, 2009.

[7] R. Baden, A. Bender, N. Spring, B. Bhattacharjee, Daniel Starin, Persona: an online social network with user-defined privacy, SIGCOMM, 2009.

[8] Oleksandr Bodriagov, Sonja Buchegger, P2p social networks with broadcast encryption protected privacy, in: Privacy and Identity Management for Life, Springer, 2012, pp. 197–206.

[9] Dan Boneh, Xavier Boyen, Eu-Jin Goh, Hierarchical identity based encryption with constant size ciphertext, in: Advances in Cryptology EUROCRYPT 2005, Lecture Notes in Computer Science, vol. 3494, Springer, Berlin Heidelberg, 2005, pp. 440–456.

[10] S. Buchegger, D. Schioberg, L. Vu, A. Datta, Peerson: P2p social networking – early experiences and insights, SNS, 2009.

[11] Meeyoung Cha, Haewoon Kwak, Pablo Rodriguez, Yong-Yeol Ahn, Sue Moon, I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system, in: IMC, ACM, 2007.

[12] A. Cutillo, R. Molva, T. Strufe, Safebook: feasibility of transitive cooperation for privacy on a decentralized social network, 2009.

[13] A. Cutillo, R. Molva, T. Strufe, Safebook: a privacy preserving online social network leveraging on real-life trust, IEEE Commun. Mag. (2009).

[14] Emiliano De Cristofaro, Mark Manulis, Bertram Poettering, Private discovery of common social contacts, Int. J. Inform. Security 12 (1) (2013) 49–65.

[15] Ccile Delerable, Identity-based broadcast encryption with constant size ciphertexts and private keys, in: Advances in Cryptology ASIACRYPT, 2007.

[16] Claudia Diaz, Seda Gürses, Understanding the landscape of privacy technologies, Extended abstract of invited talk at Information Security Summit, 2012.

[17] Peter Druschel, Antony Rowstron, Past: a large-scale, persistent peer-to-peer storage utility, in: Hot Topics in Operating Systems, 2001.

[18] M. Durr, M. Maier, F. Dorfmeister, Vegas – a secure and privacy-preserving peer-to-peer online social network, in: Privacy, Security, Risk and Trust (PASSAT), 2012, pp. 868–874.

[19] Catherine Dwyer, Starr Roxanne Hiltz, Katia Passerini, Trust and privacy concern within social networking sites: a comparison of facebook and myspace, in: AMCIS, 2007.

[20] M. Falch, A. Henten, R. Tadayoni, I. Windekilde, Business models in social networking, in: CMI International Conference – Social Networking and Communities, 2009.

[21] Antonino Famulari, Artur Hecker, Mantle: a novel dosn leveraging free storage and local software, in: ICAIT, 2012.

[22] H. Gao, J. Hu, T. Huang, J. Wang, Y. Chen, The status quo of online social network security: a survey, IEEE Int. Comput. (2011).

[23] Vipul Goyal, Omkant Pandey, Amit Sahai, Brent waters, Attribute-based encryption for fine-grained access control of encrypted data, in: CCS, 2006.

[24] K. Graffi, S. Podrajanski, P. Mukherjee, A. Kovacevic, R. Sreinmetz, A distributed platform for multimedia communities, in: Tenth IEEE International Symposium on Multimedia, 2008.

[25] Benjamin Greschbach, Sonja Buchegger, Friendly surveillance – a new adversary model for privacy in decentralized online social etworks, in: Current Issues in IT Security, 2012, pp. 195–206.

[26] Benjamin Greschbach, Gunnar Kreitz, Sonja Buchegger, The devil is in the metadata – new privacy challenges in decentralised online social networks, in: SESOC (PERCOM Workshops), 2012.

[27] Saikat Guha, Kevin Tang, Paul Francis, NOYB: privacy in online social networks, in: First Workshop on Online Social Networks, 2008.

[28] Felix Günther, Mark Manulis, Thorsten Strufe, Cryptographic treatment of private user profiles, in: Financial Cryptography and Data Security, Springer, 2012, pp. 40–54.

[29] Sonia Jahid, P Mittal, Nikita Borisov, EASiER: encryption-based access control in social networks with efficient revocation, in: ASIACCS, 2011.

[30] Sonia Jahid, Shirin Nilizadeh, Prateek Mittal, Nikita Borisov, Apu Kapadia, Decent: a decentralized architecture for enforcing privacy in online social networks, SESOC (PERCOM Workshops), 2012, pp. 326–332.

[31] David Koll, Jun Li, Xiaoming Fu, With a little help from my friends: replica placement in decentralized online social networks, 2013.

[32] M. Kryczka, R. Cuevas, C. Guerrero, A first step towards user assisted online social networks, SNS, 2010.

[33] D. Liu, A. Shakimov, R. Càceres, A. Varshavsky, L.P. Cox, Confidant: protecting osn data without locking it up, in: Middleware, 2011.

[34] Arvind Narayanan, Vincent Toubiana, Solon Barocas, Helen Nissenbaum, Dan Boneh, A critical look at decentralized personal data architectures. arXiv preprint arXiv:1202.4503, 2012.

[35] Rammohan Narendula, T.G. Papaioannou, Karl Aberer, A decentralized online social network with efficient user-driven replication, in: Enabling Technologies: Infrastructures for Collaborative Enterprises, 2010.

[36] Shirin Nilizadeh, Sonia Jahid, Prateek Mittal, Nikita Borisov, Apu Kapadia, Cachet: a decentralized architecture for privacy preserving social networking with caching, in: CoNEXT, 2012.

[37] Thomas Paul, Sonja Buchegger, Thorsten Strufe, Decentralizing social networking services, in: International Tyrrhenian Workshop on Digital Communications, 2010.

[38] Thomas Paul, Benjamin Greschbach, Sonja Buchegger, Thorsten Strufe, Exploring decentralization dimensions of social network services: adversaries and availability, in: HotSoc (KDD Workshop), 2012.

[39] T. Perrin, Public key distribution through "cryptoids", in: Workshop on New Security Paradigms, 2003, pp. 82–102.

[40] Fatemeh Raji, Ali Miri, Mohammad Davarpanah Jazi, Behzad Malek, Online social network with flexible and dynamic privacy policies, in: CSSE, IEEE, 2011, pp. 135–142.

[41] Antony Rowstron, Peter Druschel, Pastry: scalable, decentralized object location, and routing for large-scale peer-to-peer systems, Middleware, 2001.

[42] Amit Sahai, Brent Waters, Fuzzy identity-based encryption, in: Advances in Cryptology EUROCRYPT 2005, Springer, 2005.

[43] Stephan Schulz, Thorsten Strufe, d^2 deleting diaspora: practical attacks for profile discovery and deletion, in: ICC, 2013, pp. 2042–2046.

[44] Lorenz Schwittmann, Christopher Boelmann, Matthaus Wander, Torben Weis, Sonet-privacy and replication in federated online social networks, in: ICDCSW, Springer, 2013, pp. 51–57.

[45] S. Seong, J. Seo, M. Nasielsky, D. Sengupta, S. Hangal, S.K. Teh, R. Chu, B. Dodson, M.S. Lam, Prpl: a decentralized social networking infrastructure, in: Workshop of on Mobile Computing and Services: Social Networks and Beyond, 2010.

[46] Nashid Shahriar, Shihabur Rahman Chowdhury, Mahfuza Sharmin, Reaz Ahmed, Raouf Boutaba, Bertrand Mathieu, Ensuring beta-availability in p2p social networks, in: ICDCSW, IEEE, 2013, pp. 150–155.

[47] A. Shakimov, A. Varshavsky, L.P. Landon, R. Càceres, Privacy, cost, and availability tradeoffs in decentralized osns, WOSN, 2009.

[48] Rajesh Sharma, Anwitaman Datta, Supernova: super-peers based architecture for decentralized online social networks, in: COMSNETS, IEEE, 2012, pp. 1–10.

[49] Rajesh Sharma, Anwitaman Datta, M. DeH'Amico, Pietro Michiardi, An empirical study of availability in friend-to-friend storage systems, in: IEEE P2P, IEEE, 2011, pp. 348–351.

[50] Daniel Stutzbach, Reza Rejaie, Understanding churn in peer-to-peer networks, in: IMC, ACM, 2006, pp. 189–202.

[51] J. Sun, X. Zhu, Y. Fang, A privacy-preserving scheme for online social networks with efficient revocation, in: INFOCOM, 2010.

[52] Florian Tegeler, David Koll, Xiaoming Fu, Gemstone: empowering decentralized social networking with high data availability, in: GLOBECOM, IEEE, 2011, pp. 1–6.

[53] A. Tootoonchian, S. Saroiu, Y. Ganjali A. Wolman, Lockr: better privacy for social network, CoNEXT, 2009.

[54] C. Wilson, T. Steinbauer, G. Wang, A. Sala, H. Zheng, B.Y. Zhao. Privacy, availability and economics in the polaris mobile social network, in: HotMobile, 2011.

Thomas Paul is a Ph.D. Student and Research Engineer at TU-Darmstadt. He received his Dipl. Wirt.-Inf. degree from TU Ilmenau. After spending three years as a business (pricing) consultant, he returned to science. His thesis will focus on security in Online Social Networks.

Antonino Famulari is a Ph.D Student at Telecom ParisTech, affiliated with the doctoral school in Innovation and Entrepreneurship of the EIT. Previously, he obtained his MsC in telecommunication engineering (Pisa) and gained experience in consulting and research at the Italian National Institute of Nuclear Physics. His thesis will focus on protection of online privacy, in particular in Online Social Networks and content sharing platforms.

Thorsten Strufe is professor Privacy and IT Security at Technische Universität Dresden. His research interests lie in the areas of privacy and resilience, especially in the context of social networking services and large scale distributed systems. Recently, he has focused on studying user behavior and security in Online Social Networks and possibilities to provide privacy-preserving and secure social networking services and big data solutions.

He was appointed professor for Peer-to-Peer networks at Technische Universität Darmstadt, Germany, from 2009 to 2014, and visiting professor for Dependable Distributed Systems at University of Mannheim, Germany, throughout 2011. Previously, he took posts as senior researcher at EURECOM and at TU Ilmenau, working on analysis and the security of Online Social Networks, as well as resilient networking. He received his PhD degree from TU Ilmenau in 2007. His PhD thesis deals with the construction of censorship resistant and network efficient overlay topologies for live multimedia streaming, and means to making them especially resilient towards both the failure of nodes and DoS attacks.