González-Manzano L., Slaymaker M., de Fuentes J.M., Vayenas D. (2018) SoNeUCONABCPro: An Access Control Model for Social Networks with Translucent User Provenance. In: Lin X., Ghorbani A., Ren K., Zhu S., Zhang A. (eds) Security and Privacy in Communication Networks. SecureComm 2017. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol 239. Springer, Cham. DOI https://doi.org/10.1007/978-3-319-78816-6_17

$SoNeUCON_{ABC}Pro$: an access control model for social networks with translucent user provenance

Lorena González-Manzano[1] , Mark Slaymaker[2] , Jose M. de Fuentes[1] , Dimitris Vayenas[3]

[1]Carlos III University of Madrid, Leganés, Spain. {lgmanzan,jfuentes}@inf.uc3m.es
[2]The Open University, Walton Hall, Milton Keynes, UK. mark.slaymaker@open.ac.uk
[3]Oxford University Computing Laboratory, Oxford, UK.
dimitris.vayenas@exeter.ox.ac.uk

Abstract. Web-Based Social Networks (WBSNs) are used by millions of people worldwide. While WBSNs provide many benefits, privacy preservation is a concern. The management of access control can help to assure data is accessed by authorized users. However, it is critical to provide sufficient flexibility so that a rich set of conditions may be imposed by users. In this paper we coin the term *user provenance* to refer to tracing users actions to supplement the authorisation decision when users request access. For example restricting access to a particular photograph to those which have "liked" the owners profile. However, such a tracing of actions has the potential to impact the privacy of users requesting access. To mitigate this potential privacy loss the concept of *translucency* is applied. This paper extends $SoNeUCON_{ABC}$ model and presents $SoNeUCON_{ABC}Pro$, an access control model which includes translucent user provenance. Entities and access control policies along with their enforcement procedure are formally defined. The evaluation demonstrates that the system satisfies the imposed goals and supports the feasibility of this model in different scenarios.

Key words: Social networks, Access Control, User provenance, Translucency

1 Introduction

The continuing proliferation of Web-Based Social Networks (WBSNs) encourages their study and research. This escalation raises questions about security and privacy due to the amount of managed personal data being shared. For instance, each minute around 2.5 million items are shared on Facebook and 200,000 photos are uploaded to Instagram[1]. Facebook has increased the amount of privacy controls, enabling users to restrict the content that is viewable by others. Thus,

[1] http://aci.info/2014/07/12/the-data-explosion-in-2014-minute-by-minute-infographic/

when a user writes a message or adds a friend, privacy controls associated with that content will dictate what is viewable by his friends[2].

Access control has been a challenging matter [5, 7]. It is considered such an important thing that [27] considers that the management of who accesses data should be a requirement whatever the cost. An important aspect is to provide an access control mechanism that is both flexible and fine-grained. There has been previous work on *data provenance*, defined as the process of tracing and recording the origin of data and any subsequent change [3][24]. Based on this concept, we coin the term *user provenance* to refer to the process of tracing users' actions, and using that information as a basis for decisions related to granting access. User provenance would make it possible to include additional constrains on users requesting access based on, for instance, where the user comes from or the actions that the user has previously performed.

There are several different user actions in WBSNs, i.e. the addition of comments, the uploading of photos, etc. User provenance could offer interesting access control management alternatives in this respect. The following set of paradigmatic scenarios motivates the development of an access control model that addresses user provenance.

- **Customer acquisition.** Parker's, a well-known restaurant, wants to implement an aggresive marketing campaign to steal clients of competitors. Thus, access to a special promotion is granted only to customers that have *visited* the Facebook profile of competing restaurants at least once in the last week.
- **Loyalty program.** Christian loves keeping up with the latest fashions as well as receiving feedback about his new clothes. He usually uploads photos of his new clothes to Facebook and users who *make comments* on them are allowed to access additional fashion photos he has posted. In this way, Christian limits the number of photos non-interested users can access while allowing interested ones to view a more extensive range of images.
- **Focused access.** Julia went to a Bon Jovi's concert and uploaded photos of the event to Facebook. To prevent Bon Jovi's detractors to post negative comments or create mocking memes based on these photos, she decided to grant access only to actual fans – users who have *liked* Bon Jovi's contents at least five times in the month.

According to these scenarios, the potential for privacy loss cannot be disregarded in the context of user provenance. Tracing user actions means that they are potentially transparent to the other users as these actions become part of the access control process. While tailored access control is desirable, transparency can directly affect privacy. An analogy can be drawn concerning glass-walled houses in which the clear glass walls makes it easy for anybody to look inside them. A potential method of limiting this affect is applying the concept of *translucency*, introduced by Mike Leiter [19], which can be used to balance transparency and privacy [29]. Using a smoked glass-walled home will still allow an onlooker to look inside but reducing the amount of details that can be

[2] http://www.jonloomer.com/2012/05/06/history-of-facebook-changes/

ascertained. Analogously, integrating a translucency mechanism as part of the user provenance access control management is desirable. In this way, users control actions applied in the access control process and actions that should remain private.

To the best of our knowledge, no single access control model for WBSNs has been proposed enabling the expressiveness permitted by user provenance. $SoNeUCON_{ABC}$ [15] already considers the needs of flexibility, fine-granularity, attribute management and usage control, which are desirable access control properties. In this paper, an extension called $SoNeUCON_{ABC}Pro$ is proposed to address translucent user provenance. The behaviour of users is considered in the access control enforcement process through the management of performed actions but also considering the users right to keep some actions hidden.

The structure of the paper is as follows. Section 2 briefly introduces $SoNeUCON_{ABC}$. The proposed model is presented in Section 3. The definition and enforcement of access control policies are described in Section 4, with Section 5 providing an evaluation. An overview of other related work is described in Section 6. Finally, in Section 7 conclusions and future work are outlined.

2 Background

In this Section $SoNeUCON_{ABC}$ access control model [15] is introduced. $SoNeUCON_{ABC}$ is an expressive usage control model that manages six WBSN features, namely, common-contacts, clique, distance, multi-path, direction and flexible attributes [13][8][5][4].

$SoNeUCON_{ABC}$ is composed of seven elements: *Subjects* (S) together with *Subject attributes* ($ATT(S)$) refer to WBSN users and their attributes; *Objects* (O) together with *Object attributes* ($ATT(O)$) correspond to WBSN data and their attribute; and *Relationships* (RT) together with *Relationship attributes* ($ATT(RT)$) refer to the set of relations and attributes that exist between a pair of users, with direct relationships denoted as E and $ATT(E)$ their attached attributes; *Rights* (R) correspond to actions that can be performed over objects O; *Authorizations* (A) are rules to be satisfied to grant a subject a right on an object; *Obligations* (B) correspond to requirements to be met before or during the usage process; and *Conditions* (C) are requirements needed regarding the context features, eg. network availability.

In $SoNeUCON_{ABC}$, access control policies (ρ) consist of subjects, objects, relationships predicates (ρ_s, ρ_o and ρ_{rt} respectively), a right (r) is also provided as well as any obligations (∂_b) and conditions (∂_c) to be satisfied. An access control policy ρ is expressed as $\rho(\rho_s; \rho_o; \rho_{rt}; r; \partial_b; \partial_c)$.

An example of an access control policy is presented: *Access is granted to photos entitled "Party" to friends of a friend if they are under 30 years old or if they are under 25 years and have studied computer science.*

$\rho = (((age < 30) \vee ((age < 25) \wedge (studies = c.science)));\ (title = party);\ ((((role = friend);\ (role = friend))), \emptyset, \emptyset);\ read;\ \emptyset;\ \emptyset)$

Note that symbol $\emptyset$ is applied for policy elements which do not need to be involved in the access control management process. The first $\emptyset$ means that multiple paths are not managed, the second one that cliques between users are not considered and the final pair of $\emptyset$ means that conditions and obligations respectively are not included in this policy. See [15] for details.

3 $SoNeUCON_{ABC}Pro$ proposal

This section outlines the main features of $SoNeUCON_{ABC}Pro$. For the ease of reading Table 1 presents main used notation.

Table 1. Notation table

ρ	Access control policy
ρ_t	Translucency policy
s_i	A subject i
o_i	An object i
rt_i	A relationship i
r	Right to be granted
∂_c	Conditions
χ	Obligations
ξ	User provenance pred.
∂_o	Obligation different from ξ
$P_{ac_{u_i}}$	Path, actions carried out by user u_i
u_i	user i
$ac_{j-u_i:o_k}$	Action j performed by u_i over o_k
e_i	edge/ relationship i

3.1 Goals

$SoNeUCON_{ABC}Pro$ should include the management of **user provenance**, facilitating access control management that is based on the actions performed by WBSN users (called requesters) over other users' data. The system must allow the definition of access control policies that consider previous actions of requesters. The system must also enable **translucency**, allowing requesters to hide some, or all of their, actions when an access control policy is evaluated.

3.2 Supporting example

This example presents actions carried out by Daniel when he interacts in a WBSN with Alice, Bob and Charly. According to what is described here, access control has to be managed considering interactions performed by Daniel. Moreover, all restrictions, either performed by Daniel or by other user, as it is the case of Bob, have to be excluded in the access control process to respect users' privacy.

In a WBSN Daniel interacts with his direct friends Alice, Bob and Charly. Fig. 1 depicts the interactions that Daniel made between the 1st and 5th of

June. On the 1st June Daniel added a like to photo1, photo2 and to the profile of Charly. In addition Daniel posted a comment on Alice's wall. Over the four days of activity covered by Fig. 1, Daniel performed a total of 11 actions on various elements of the profiles of his contacts. Moreover, Daniel wants to hide that he has clicked "like" in any of his friends' profiles, Alice's and Charly's profiles in this scenario.

Additionally, Bob specifies that access to photos entitled "SummerWithAlice" would be only granted to WBSN users who like Alice's profile.

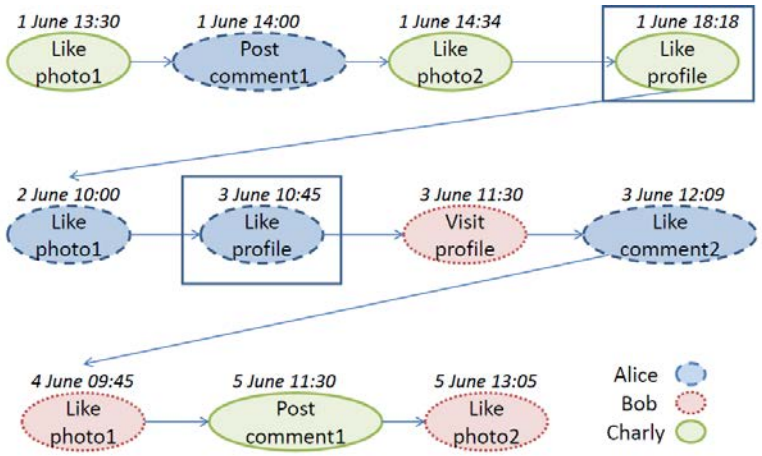

Fig. 1. Actions of Daniel over his contacts' data

3.3 Model definition

Expressing user provenance in terms of $SoNeUCON_{ABC}$ could be modelled as some Rights r_i that are given after fulfilling some Obligations b_i. Let us consider the supporting example, the Right of accessing photos entitled "SummerWithAlice" is given to users that have liked Alice's profile (Obligation). However, Obligations in $SoNeUCON_{ABC}$ cannot be related to specific objects or subjects. The proposed case needs to express that the obligation is to access the profile (object) of Alice (subject). This lack of expressiveness motivates the extension presented herein.

User provenance management requires including performed actions within the access control process. WBSN actions, defined as *Actions AC*, are modelled as a particular type of *Obligation* b_i. Thus, $SoNeUCON_{ABC}Pro$ extends $SoNeUCON_{ABC}$ including entity *Actions AC* within *Obligations B* together with attached links (Fig. 2). *AC* are performed by subjects S over objects O and then, *AC* is related to S and O. Consequently, including *AC* in B comprises new links whose management needs to be specified.

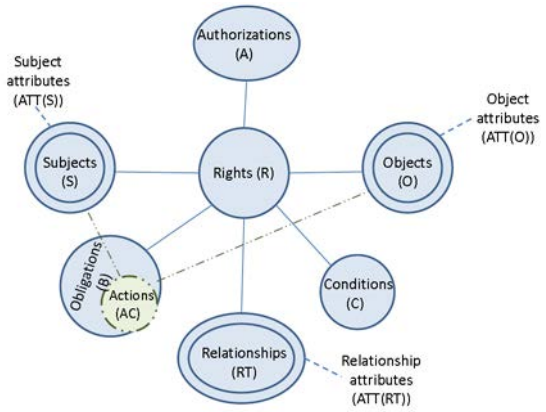

Fig. 2. $SoNeUCON_{ABC}Pro$

3.4 Translucent user provenance

Actions management involves the creation of a path per user u_i ($P_{ac_{u_i}}$) where nodes are WBSN actions ($ac_{j-u_i:o_k}$) performed over objects o_k that include the date and time when they are performed; and edges are time relationships (e_{ti}) among nodes. The construction of $P_{ac_{u_i}}$ comprises two steps:

1. Identification of all users U' that have resources over which the requester, a user u_i, has performed an action.
2. Ordering of actions based on date and time. Note that sequential actions over the same object are represented as different nodes connected in temporal order.

Concerning the supporting example a path is constructed based on the date and time of actions performed by Daniel. Fig. 3 depicts the formal representation ($P_{ac_{u_4}}$), being Daniel u_4 and Alice, Bob and Charly u_1, u_2 and u_3 respectively. When Daniel, u_4, requests a permission over an object of other user, access control involves verifying some actions of the created path to grant or deny the requested permission accordingly. If Daniel would not mind to disclose any action and if he requests access to "SummerWithAlice" photos, the created path is evaluated and the access granted because he clicked "like" on Alice's, u_1, profile on June 3rd ($ac_{6-u_1:o_6}$).

$SoNeUCON_{ABC}Pro$ also manages translucency. Given the inclusion of AC within B, translucency is based on managing which ac_i performed by the requester u_i over an o_i of a user u_j should remain accessible. In other words, access to chosen nodes $ac_{j-u_i:o_k}$ is denied such that $ac_{j-u_i:o_k}$ and attached e_{ti} are deleted from $P_{ac_{u_i}}$ in the access control enforcement process. Recalling the supporting example, as Daniel, u_4, does not want to disclose that he has liked Alice's, u_1, and Charly's, u_3, profiles, actions $ac_{4-u_3:o_4}$ and $ac_{6-u_1:o_6}$ (highlighted in Fig. 3) are not involved in the process. Then, "SummerWithAlice" pictures

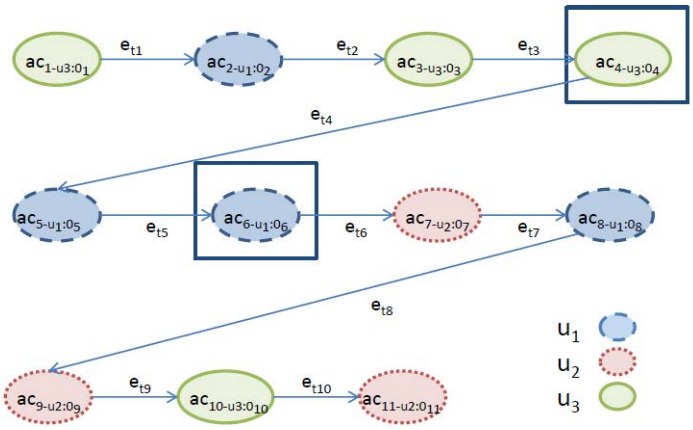

Fig. 3. $P_{ac_{u_4}}$ formal representation of supporting example (Fig. 1)

are denied to Daniel, u_4. More specifically, $ac_{3-u_3:o_3}$ would be linked to $ac_{5-u_1:o_5}$ through e_{t3} and $ac_{5-u_1:o_5}$ would be linked to $ac_{7-u_2:o_7}$ through e_{t5}.

4 Access control policies

This Section includes the description and enforcement of access control policies.

4.1 Description

Access control policies are enhanced to address user provenance and translucency. While the former requires the update of $SoNeUCON_{ABC}$ policies, translucency management requires the inclusion of a new set of policies called ***translucency policies*** (ρ_t).

Concerning user provenance, the same operators and attributes as those applied in $SoNeUCON_{ABC}$ [15] are considered. Nonetheless, access control policies are defined in terms of $ATT(S)$, $ATT(O)$, $ATT(RT)$, R, C, B and AC. In particular, an access control policy is formally defined as $\rho(\rho_s; \rho_o; \rho_{rt}; r; \chi; \partial_c)$.

Recalling Section 2, the only difference is that χ replaces ∂_b. In fact, χ is a superset containing $SoNeUCON_{ABC}$ obligations as well as the user provenance actions (ξ) introduced in $SoNeUCON_{ABC}Pro$. χ is described as follows using BNF notation [28]:

- $\chi ::= (\emptyset|\xi^*|\partial_b) ::= (\emptyset|(act_i; dt; \rho'_w)^*|\partial_b) ::= (\emptyset|(act_i; dt; \rho'_s; \rho'_o; \rho'_{rt})^*|\partial_b)$
  - ξ comprises predicates applied for user provenance management.
  - ∂_b refers to any type of obligation different from those related to user provenance, e.g. the need to have 10 contacts at least. This type of obligation is analogous to the ones presented in $SoNeUCON_{ABC}$.

- act_i refers to all possible actions that can be applied in the WBSN context. For instance, Liked, Visited, Commented, etc. Note that xAPI can be used to represent user actions [1].
- dt refers to the date (d) in the form YYYY/MM/DD and the time (t) in the form HH:MM:SS when act_i is performed. Data and time follow ISO 8601 [30]. Any element, e.g. DD, can take symbol $*$ meaning that no restrictions are established. dt can take symbol $\emptyset$ meaning that no element has restrictions.
- ρ'_w refer to subjects, objects and relationship predicates, where w can take three possible values – s, o or e –, to indicate its relation to subjects, objects or relationships. Specifically, ρ'_s refers to subjects with a relationship type ρ'_{rt} with the owner of the requested object who perform r_i over objects linked to ρ'_o.

Example. Recalling the supporting example, the following user provenance policy expresses that access to "SummerWithAlice" photos is only granted to users who like Alice's, u_1, profile, χ $(Liked; */*/*-* : * : *; (name = Alice);\ (title = profile); ((((\emptyset))),\ \emptyset,\ \emptyset))$.

On the other hand, ρ_t are proposed to limit which actions are applied in the access control process and which ones remain hidden. All WBSN actions are accessible to all users by default – the whole $p_{ac_{u_i}}$ is applied in managing access control. However, if ρ_t exist, they are firstly evaluated against $P_{ac_{u_i}}$. Actions $ac_{j-u_i:o_k}$ which satisfy established ρ_t are removed from the graph. ρ_t are formally described as follows again applying BNF notation [28].

- $\rho_t ::= (act_i; dt; \rho''_w) ::= (act_i; dt; \rho''_s; \rho''_o; \rho''_{rt})$
  - act_i refers to WBSN actions performed over objects linked to ρ''_o.
  - dt refers to the date and time when act_i is performed. Its structure is analogous to the one presented in χ.
  - ρ''_w again involves subject, object and relationship predicates (ρ''_s, ρ''_o and ρ''_{rt} respectively) but meaning that removed nodes are those where objects are linked to ρ''_o whose owner satisfies ρ''_s and has a relationship ρ''_{rt} with the data requester.

Example. Considering Fig. 3 and the supporting example, Daniel, u_4, does not want to disclose that he has liked his direct friends' profiles. Thus, he establishes $\rho_{t_1}(Like; \emptyset;\ (title = profile); ((((role = friend))),\ \emptyset,\ 1))$, such that the resulting $P_{ac_{u_4}}$ would be the one depicted in Fig. 3 removing nodes within rectangles. In this way, Daniel, u_4 limits which actions are accessible becoming translucent.

4.2 Policy enforcement

$P_{ac_{u_i}}$ is defined as an ordered list ($lpath_{u_i}$) where each position is an action $ac_{j-u_i:o_k}$ together with attached object o_k. $lpath_{u_i}$ is formally represented as $lpath_{u_i}\{o_t, ac_{k-u_i:o_t}; o_{t+1}, ac_{(k+1)-u_i:o_{k+1}}; ...\}$. For instance, based on Fig. 3, $lpath_{u_4}$ $\{o_1, ac_{1-u_3:o_1}; o_2, ac_{2-u_1:o_2}; o_3, ac_{3-u_3:o_3}; o_4, ac_{4-u_3:o_4}; .....; o_{11}, ac_{11-u_2:o_{11}}\}$.

After the construction of $P_{ac_{u_i}}$ per access request (s, o, r) (where s is the requester, o the requested object and r the requested right over o), all access control policies ρ and translucency policies ρ_t are evaluated. ρ_t are firstly evaluated against *lpath*. Function *evaluateTransPolicies* is executed with the inputs of ρ_t, *lpath* and *req*, where *req* is a reference to data pertaining to the requester, such as his objects and relationships. If the result of the evaluation is 'true' the appropriate elements of *lpath* are removed and thus, *newlpath* is created and applied in the evaluation of ρ. Pseudo-code of *evaluateTransPolicies* is depicted in Algorithm 1 where functions *Match*, *MatchRT*, *GetSubAtt*, *GetObjAtt*, *CreateRT* and *GetAdmin* are developed in $SoNeUCON_{ABC}$ (see [15] for details). These functions are used to verify that objects O, subjects S and relationships RT predicates ρ''_w involved in ρ_t match O, S and RT involved in *lpaths*. Note that symbol "." is used to access the content of an element and the expression $list[pos]$ refers to accessing the element of *list* located at position *pos*.

Algorithm 1 evaluateTransPolicies

```
1: procedure EVALUATETRANSPOLICIES(ρ_t, lpath, req )
2:     for lpath_{u_i} ← (i = 1) to sizeOf(lpath) do
3:         if ρ_t.act_i = lpath_{u_i}[j] then
4:             if verifyDateTime(ρ_t.dt, lpath_{u_i}[j]) then
5:                 adminLpath = GetAdmin(lpath_{u_i}[j].o)
6:                 attSubj = GetSubAtt(adminLpath, lpath_{u_i}[j].ρ_s)
7:             end if
8:             if Match(attSubj, ρ_t.ρ''_s) then
9:                 attObj = GetObjAtt(lpath_{u_i}[j].o, lpath_{u_i}[j].ρ_o)
10:            end if
11:            if Match(attObj, ρ_t.ρ''_o) then
12:                rt = CreateRT(adminLpath, req, 1)
13:            end if
14:            if MatchRT(ρ_t.ρ''_r t, rt, u_i, 1) then
15:                return lpath_{u_i} node marked as not usable. newlpath
16:            end if
17:        end if
18:    end for
19: end procedure
```

Subsequently, access control policies ρ are evaluated. The evaluation of predicates ρ_o, ρ_s and ρ_{rt}, conditions ∂_c and the subset of obligations ∂_b are analogous to the corresponding elements of $SoNeUCON_{ABC}$ [15]. Then, the evaluation of χ is what needs to be described herein (Function *evaluate*χ, Algorithm 2). It is similar to *evaluateTransPolicies*, the only difference is when all $\xi \in \chi$ are evaluated over *newlpath*. If the result is 'true' for all ξ, the requested r_i over o_i is granted whether results of evaluating the remaining elements in ρ are also 'true'.

Algorithm 2 evaluateχ

```
1: procedure EVALUATEχ(ρ_r, newlpath, req )
2:     for χ.ξ[h] ← (h = 1) to sizeOf(χ) do
3:         for newlpath_{u_i} ← (i = 1) to sizeOf(newlpath) do
4:             if χ.ξ[h].act_i = newlpath_{u_i}[j] then
5:                 if verifyDateTime(χ ξ[h].dt, newlpath_{u_i}[j]) then
6:                     adminLpath = GetAdmin(newlpath_{u_i}[j].o)
7:                     AttSubj = GetSubAtt(adminLpath, newlpath_{u_i}[j].ρ_s)
8:                 end if
9:                 if Match(attSubj, χ ξ[h].ρ''_s) then
10:                    AttObj = GetObjAtt(newlpath_{u_i}[j].o, newlpath_{u_i}[j].ρ_o)
11:                end if
12:                if Match(AttObj, χ.ξ[h].ρ''_o) then
13:                    rt = CreateRT(adminLpath, req, 1)
14:                end if
15:                if MatchRT(χ.ξ[h].ρ''_r t, rt, u_i, 1)) then
16:                    return χ verified. Result true
17:                end if
18:            end if
19:        end for
20:    end for
21: end procedure
```

5 Evaluation

The evaluation of $SoNeUCON_{ABC}Pro$ comprises a goals analysis and a temporal workload assessment.

5.1 Goals analysis

$SoNeUCON_{ABC}Pro$ addresses user provenance together with translucency. The former feature is achieved by the inclusion of actions within obligations together with management issues, namely the update of access control policies and the enforcement procedure. Translucency is achieved by creating and managing policies by which users only disclose chosen actions.

Note that to apply $SoNeUCON_{ABC}Pro$ in a real WBSN the following three guidelines should be considered: 1) WBSNs should allow the establishment of ρ and ρ_t; 2) attributes within ρ such as age, role, etc. which are already used and stored by WBSNs, should be involved in the access control process; and 3) user actions have to be recorded by WBSNs and those actions included in ρ_t removed from the access control enforcement process.

5.2 Temporal workload assessment

In $SoNeUCON_{ABC}Pro$ access control management is based on the evaluation of policies ρ and translucency policies ρ_t per user request. A critical aspect is

to keep the temporal workload under usability limits. In this regard, ρ_t will be managed off-line whereas ρ are managed on-line. Each part will be analyzed separately.

Experimental settings $SoNeUCON_{ABC}Pro$ is devoted to the very same goal as its ancestor, $SoNeUCON_{ABC}$ – access control. Therefore, experiments are focused on measuring how much time it takes to assess the policies at stake in different social network scenarios. The settings mainly relate to three aspects – the social networks, the policies and the computational resources. Regarding the first aspect, the same four WBSNs created in the evaluation of $SoNeUCON_{ABC}$ have been considered herein. For illustration purposes, Table 2 depicts the number of nodes ($\#v_i$), relationships ($\#e_i$) and relationships per node ($\overline{e_i/v_i}$) of proposed WBSNs.

Table 2. WBSNs structure

WBSNs id	$\#e_i$	$\#v_i$	$\overline{e_i/v_i}$
1	2,980,388	50,000	60
2	5,965,777	50,000	120
3	8,949,375	50,000	185
4	10,929,713	50,000	219

As $SoNeUCON_{ABC}$ policies did not consider user provenance or translucency, they could not be directly applied to assess $SoNeUCON_{ABC}Pro$. In this case we consider that common actions that users perform in a WBSN such as Facebook are *Liked*, *Photos uploaded*, *Sent messages*, *Shared items* and *Comments*, where the percentage of actions usage is the one presented in Table 3. However, apart from actions, the elements involved in policies of $SoNeUCON_{ABC}$ are similar to those involved in user provenance and translucency – they affect subjects, objects and relationships. Thus, we assume that these policies have similar computational requirements than user provenance or translucency ones. For simplicity we keep the same policies than $SoNeUCON_{ABC}$ (see [15]).

Table 3. Percentage of actions usage in Facebook [3] [4]

Likes	Photos uploaded	Sent messages	Shared items	Comments
43.75	0.30	9.72	46.18	0.00071

The experiments were carried out on a Intel Core Due E8400 3.2GHz processor with 4GB of RAM and Ubuntu 12.04. This experiment is designed to act as a crude proof of principle as it is running on a modest system.

[3] `http://blog.wishpond.com/post/115675435109/40-up-to-date-facebook-facts-and-stats` , last access June 2017

[4] `https://zephoria.com/top-15-valuable-facebook-statistics` , last access June 2017

Off-line part: Translucency The evaluation of ρ_t consists of creating $P_{ac_{u_i}}$ per user u_i and enforcing the verification of ρ_t over such path. Most WBSNs store a timeline of our activities[5]. Then, $P_{ac_{u_i}}$ can be created at runtime avoiding the cost of its creation when policy enforcement is carried out. Likewise, the separation of actions could benefit performance to a great extent, for instance, creating a path $P_{ac_{j,u_i}}$ for each type of action j. In the same way, the evaluation of ρ_t over each $P_{ac_{j,u_i}}$ can be carried out off-line also benefiting performance, that is after the execution of a set of actions instead of per user's request. This simplifies the implementation of translucent user provenance in a real environment.

This workload is measured as follows. For each action in $P_{ac_{u_i}}$, it is necessary to evaluate if conditions are met (thus hiding the action from access control evaluation) or not. As aforementioned, this evaluation involves the same elements as those existing in $SoNeUCON_{ABC}$ policies and then, in this previous model the evaluation of proposed policies takes 13 ms at minimum and 184.5 ms on average (see Appendix for details). Therefore, we take these values as the expected time to assess each action.

Regarding the amount of actions (i.e. the length of $P_{ac_{j,u_i}}$), we propose different scenarios based on the amount of contacts and the number of actions over each contact's data. Particularly, we consider 25, 50, 100, 300 contacts and 10, 25, 50, 75, 300, 450,750,1000,10000 actions per contact. Thus, $P_{ac_{j,u_i}}$ ranges from 250 to 3000000 actions, though for performance reasons different paths per type of action could be distinguished. Note that the amount of contacts is in line with current figures, as 338 users is the average amount of Facebook friends[6].

Considering established parameters, temporal workload of evaluating translucency policies ρ_t is presented in Table 4. Depending on the type of action within ρ_t, the temporal workload is highly affected because the higher the usage of actions (recall Table 3), the higher the nodes in $P_{ac_{u_i}}$ to evaluate. Though results are better when actions *Photos uploaded* or *Comments* are involved in ρ_t, as this process is performed off-line, the impact of temporal workload is not a big issue. For instance, when the action type is *Comments*, the evaluation takes 3 ms for 10 actions and around 0.39 ms for 1000 actions and 300 contacts in the average case. However, when other actions are at stake, i.e. *Shared items*, the evaluation takes 1.4 min for 10 actions and 42 min for 300 actions and 100 contacts in the average case.

On-line part: Access control with user provenance The evaluation of ρ that include user provenance is carried out on-line. As opposed to $SoNeUCON_{ABC}$ assessment, in this proposal obligations B are critical – recall that user provenance can be seen as obligations involving actions, $\xi \in \chi$.

Policy ρ enforcement can be divided into two main parts. First, the evaluation of predicates regarding the object (ρ_o), the subject (ρ_s) and its relationships

[5] https://es-la.facebook.com/notes/radio-949/timeline/309814275719798/ , last access June 2017

[6] http://www.pewresearch.org/fact-tank/2014/02/03/6-new-facts-about-facebook/ , last access June 2017

Table 4. Off-line part: translucency assessment. Temporal workload (sc)

		# of actions per user's data								
		10	25	50	75	300	450	750	1000	100000
Like										
Best time										
# contacts per user	25	1.42	4.98	12.09	14.93	42.66	63.99	106.66	142.21	14220.92
	50	2.84	9.95	24.17	29.86	85.32	127.98	213.31	284.41	28440.94
	100	5.69	19.91	48.35	59.72	170.64	255.96	426.61	568.81	56881.08
	300	17.06	59.72	145.04	179.17	511.92	767.89	1279.81	1706.41	170641.24
Average time										
# contacts per user	25	20.13	60.38	140.88	181.14	603.79	905.69	1509.48	2012.64	201263.71
	50	40.25	120.76	281.77	362.27	1207.58	1811.37	3018.95	4025.27	402526.75
	100	80.51	241.52	563.54	724.55	2415.16	3622.74	6037.90	8050.53	805052.83
	300	241.55	724.55	1690.61	2173.64	7246.47	10869.71	18116.18	24154.90	2415490.00
Photos uploaded										
Best time										
# contacts per user	25	0.01	0.04	0.09	0.12	0.33	0.50	0.83	1.11	110.61
	50	0.02	0.08	0.19	0.23	0.66	1.00	1.66	2.21	221.21
	100	0.04	0.15	0.38	0.46	1.33	1.99	3.32	4.42	442.41
	300	0.13	0.46	1.13	1.39	3.98	5.97	9.95	13.27	1327.21
Average time										
# contacts per user	25	0.16	0.47	1.10	1.41	4.70	7.04	11.74	15.65	1565.38
	50	0.31	0.94	2.19	2.82	9.39	14.09	23.48	31.31	3130.76
	100	0.63	1.88	4.38	5.64	18.78	28.18	46.96	62.62	6261.51
	300	1.88	5.64	13.15	16.91	56.36	84.54	140.90	187.87	18787.10
Sent messages										
Best time										
# contacts per user	25	0.32	1.11	2.69	3.32	9.48	14.22	23.70	31.60	3160.20
	50	0.63	2.21	5.37	6.64	18.96	28.44	47.40	63.20	6320.21
	100	1.26	4.42	10.74	13.27	37.92	56.88	94.80	126.40	12640.24
	300	3.79	13.27	32.23	39.82	113.76	170.64	284.40	379.20	37920.27
Average time										
# contacts per user	25	4.47	13.42	31.31	40.25	134.18	201.26	335.44	447.25	44725.27
	50	8.95	26.84	62.62	80.51	268.35	402.53	670.88	894.50	89450.39
	100	17.89	53.67	125.23	161.01	536.70	805.05	1341.75	1789.01	178900.63
	300	53.68	161.01	375.69	483.03	1610.33	2415.49	4025.82	5367.76	536775.56
Shared items										
Best time										
# contacts per user	25	1.50	5.25	12.76	15.76	45.03	67.55	112.58	150.11	15010.97
	50	3.00	10.51	25.52	31.52	90.06	135.09	225.16	300.21	30020.99
	100	6.00	21.01	51.03	63.04	180.12	270.19	450.31	600.41	60041.14
	300	18.01	63.04	153.10	189.12	540.36	810.55	1350.91	1801.21	180121.31
Average time										
# contacts per user	25	21.24	63.73	148.71	191.20	637.34	956.00	1593.34	2124.45	212445.03
	50	42.49	127.47	297.42	382.40	1274.67	1912.00	3186.67	4248.89	424889.35
	100	84.98	254.93	594.84	764.80	2549.33	3824.00	6373.33	8497.78	849777.98
	300	254.97	764.80	1784.53	2294.40	7649.05	11473.58	19122.63	25496.84	2549683.89
Comments										
Best time										
# contacts per user	25	$2{\cdot}10^{-5}$	$8{\cdot}10^{-5}$	$2{\cdot}10^{-4}$	$2{\cdot}10^{-4}$	$7{\cdot}10^{-4}$	$1{\cdot}10^{-3}$	$1{\cdot}10^{-3}$	$2{\cdot}10^{-3}$	0.23
	50	$5{\cdot}10^{-5}$	$1{\cdot}10^{-4}$	$3{\cdot}10^{-4}$	$4{\cdot}10^{-4}$	$1{\cdot}10^{-3}$	$2{\cdot}10^{-3}$	$3{\cdot}10^{-3}$	$4{\cdot}10^{-3}$	0.46
	100	$9{\cdot}10^{-5}$	$3{\cdot}10^{-4}$	$7{\cdot}10^{-4}$	$9{\cdot}10^{-4}$	$2{\cdot}10^{-3}$	$4{\cdot}10^{-3}$	$6{\cdot}10^{-3}$	$9{\cdot}10^{-3}$	9.29
	300	$2{\cdot}10^{-4}$	$9{\cdot}10^{-4}$	$2{\cdot}10^{-3}$	$2{\cdot}10^{-3}$	$8{\cdot}10^{-3}$	0.01	0.02	0.02	2.78
Average time										
# contacts per user	25	$3{\cdot}10^{-4}$	$9{\cdot}10^{-4}$	$2{\cdot}10^{-3}$	$2{\cdot}10^{-3}$	$9{\cdot}10^{-4}$	0.01	0.02	0.03	3.28
	50	$6{\cdot}10^{-4}$	$1{\cdot}10^{-3}$	$4{\cdot}10^{-3}$	$5{\cdot}10^{-3}$	0.01	0.02	0.04	0.06	6.56
	100	$1{\cdot}10^{-3}$	$3{\cdot}10^{-3}$	$9{\cdot}10^{-3}$	0.01	0.03	0.05	0.09	0.13	13.13
	300	$3{\cdot}10^{-3}$	0.01	0.02	0.03	0.11	0.17	0.29	0.39	39.42

(ρ_{rt}). For this part the temporal workload is exactly the time of $SoNeUCON_{ABC}$ policies. Second, the evaluation of the obligations B that the requester needs to fulfill. In this second part, we consider policies with a single user provenance obligation ξ. Given that this obligation involves the same elements that the first part (i.e. subjects, objects and relationships), we assume that it takes the same time – 13 ms (best case) and 184.5 ms (on average) – per element in the path.

Table 5 shows the time taken for the evaluation of ρ. The time needed is practically the same as the one required for translucency. The rationale behind this is that in user provenance we need to add the time for assessing the related predicates ρ_o, ρ_s and ρ_{rt} which turns out to be small as compared to the time to evaluate obligations ξ.

Despite of the similarity, the acceptance criterion for these times involves usability aspects because this is an on-line evaluation. Results are suitable if

they do not negatively affect to the user experience. Establishing 15 sc as the maximum threshold for keeping users attention[7] [22], values in bold on Table 5 are suitable. Only when types of actions *Photos uploaded* and *Comments* are involved within B the temporal workload remains within the established limit, as well as a significant amount of actions can be considered, i.e. 10000 actions for *Comments* in the average case. Nevertheless, for the remaining types of actions these results are subject to improvement, as discussed below.

Table 5. On-line part: user provenance assessment. Temporal workload (sc)

		# of actions per user's data								
		10	25	50	75	300	450	750	1000	100000
Like										
		Best time								
# contacts per user	25	**1.43**	**4.98**	**12.09**	14.95	42.83	64.25	107.08	142.78	14277.80
	50	**2.85**	**9.96**	24.18	29.88	85.49	128.24	213.73	284.98	28497.82
	100	**5.69**	19.91	48.35	59.74	170.81	256.22	427.03	569.38	56937.96
	300	17.07	59.73	145.05	179.18	512.09	768.14	1280.24	1706.98	170698.12
		Average time								
# contacts per user	25	20.21	60.46	140.96	181.38	606.21	909.32	1515.53	2020.71	202070.95
	50	40.33	120.84	281.85	362.52	1210.00	1815.00	3025.00	4033.34	403333.99
	100	80.59	241.60	563.62	724.79	2417.58	3626.37	6043.95	8058.60	805860.06
	300	241.63	724.63	1690.69	2173.88	7248.89	10873.34	18122.23	24162.97	2416297.23
Photos uploaded										
		Best time								
# contacts per user	25	**0.01**	**0.04**	**0.09**	**0.12**	**0.33**	**0.50**	**0.83**	**1.11**	111.05
	50	**0.02**	**0.08**	**0.19**	**0.23**	**0.66**	**1.00**	**1.66**	**2.22**	221.65
	100	**0.04**	**0.15**	**0.38**	**0.46**	**1.33**	**1.99**	**3.32**	**4.43**	442.85
	300	**0.13**	**0.46**	**1.13**	**1.39**	**3.98**	**5.97**	**9.96**	**13.28**	1327.65
		Average time								
# contacts per user	25	**0.16**	**0.47**	**1.10**	**1.41**	**4.71**	**7.07**	**11.79**	15.72	1571.66
	50	**0.31**	**0.94**	**2.19**	**2.82**	**9.41**	**14.12**	23.53	31.37	3137.03
	100	**0.63**	**1.88**	**4.38**	**5.64**	18.80	28.21	47.01	62.68	6267.79
	300	**1.88**	**5.64**	**13.15**	16.91	56.38	84.57	140.95	187.93	18793.38
Sent messages										
		Best time								
# contacts per user	25	**0.32**	**1.11**	**2.69**	**3.32**	**9.52**	**14.28**	23.80	31.73	3172.84
	50	**0.63**	**2.21**	**5.37**	**6.64**	19.00	28.50	47.50	63.33	6332.85
	100	**1.27**	**4.43**	**10.75**	**13.28**	37.96	56.94	94.90	126.53	12652.88
	300	**3.79**	**13.27**	32.23	39.82	113.80	170.70	284.50	379.33	37932.91
		Average time								
# contacts per user	25	**4.49**	**13.44**	31.33	40.31	134.71	202.07	336.78	449.05	44904.66
	50	**8.96**	26.85	62.63	80.56	268.89	403.33	672.22	896.30	89629.77
	100	17.91	53.69	125.25	161.06	537.24	805.86	1343.10	1790.80	179080.01
	300	53.70	161.03	375.71	483.08	1610.86	2416.30	4027.16	5369.55	536954.94
Shared items										
		Best time								
# contacts per user	25	**1.51**	**5.26**	**12.76**	15.78	45.21	67.82	113.03	150.71	15071.01
	50	**3.01**	**10.51**	25.52	31.54	90.24	135.36	225.61	300.81	30081.03
	100	**6.01**	21.02	51.04	63.06	180.30	270.46	450.76	601.01	60101.18
	300	18.02	63.05	153.10	189.14	540.54	810.82	1351.36	1801.81	180181.34
		Average time								
# contacts per user	25	21.33	63.82	148.80	191.46	639.89	959.84	1599.73	2132.97	213297.11
	50	42.57	127.55	297.51	382.66	1277.22	1915.84	3193.06	4257.41	425741.43
	100	85.06	255.02	594.93	765.05	2551.89	3827.84	6379.73	8506.30	850630.07
	300	255.05	764.88	1784.61	2294.65	7651.61	11477.41	19129.02	25505.36	2550535.97
Comments										
		Best time								
# contacts per user	25	$\mathbf{2}{\cdot}10^{5}$	$\mathbf{8}{\cdot}10^{5}$	$\mathbf{2}{\cdot}10^{4}$	$\mathbf{2}{\cdot}10^{4}$	$\mathbf{7}{\cdot}10^{4}$	$\mathbf{1}{\cdot}10^{3}$	$\mathbf{1}{\cdot}10^{3}$	$\mathbf{2}{\cdot}10^{3}$	**0.23**
	50	$\mathbf{5}{\cdot}10^{5}$	$\mathbf{1}{\cdot}10^{4}$	$\mathbf{3}{\cdot}10^{4}$	$\mathbf{4}{\cdot}10^{4}$	$\mathbf{1}{\cdot}10^{3}$	$\mathbf{2}{\cdot}10^{3}$	$\mathbf{3}{\cdot}10^{3}$	$\mathbf{4}{\cdot}10^{3}$	**0.46**
	100	$\mathbf{9}{\cdot}10^{5}$	$\mathbf{3}{\cdot}10^{4}$	$\mathbf{7}{\cdot}10^{4}$	$\mathbf{9}{\cdot}10^{4}$	$\mathbf{2}{\cdot}10^{3}$	$\mathbf{4}{\cdot}10^{3}$	$\mathbf{6}{\cdot}10^{3}$	$\mathbf{9}{\cdot}10^{3}$	**9.29**
	300	$\mathbf{2}{\cdot}10^{4}$	$\mathbf{9}{\cdot}10^{4}$	$\mathbf{2}{\cdot}10^{3}$	$\mathbf{2}{\cdot}10^{3}$	$\mathbf{8}{\cdot}10^{3}$	**0.01**	**0.02**	**0.02**	**2.78**
		Average time								
# contacts per user	25	$\mathbf{3}{\cdot}10^{4}$	$\mathbf{9}{\cdot}10^{4}$	$\mathbf{2}{\cdot}10^{3}$	$\mathbf{2}{\cdot}10^{3}$	$\mathbf{9}{\cdot}10^{4}$	**0.01**	**0.02**	**0.03**	**3.29**
	50	$\mathbf{6}{\cdot}10^{4}$	$\mathbf{1}{\cdot}10^{3}$	$\mathbf{4}{\cdot}10^{3}$	$\mathbf{5}{\cdot}10^{3}$	**0.01**	**0.02**	**0.04**	**0.06**	**6.58**
	100	$\mathbf{1}{\cdot}10^{3}$	$\mathbf{3}{\cdot}10^{3}$	$\mathbf{9}{\cdot}10^{3}$	**0.01**	**0.03**	**0.05**	**0.09**	**0.13**	**13.15**
	300	$\mathbf{3}{\cdot}10^{3}$	**0.01**	**0.02**	**0.03**	**0.11**	**0.17**	**0.29**	**0.39**	39.43

Discussion Results achieved in the experiments lead to different considerations:

[7] Although 2 sc would be a desirable threshold [22], we believe that the proposed limit is illustrative enough as it is the maximum acceptable upper limit.

- Regarding translucency, it can be performed off-line (thus not affecting usability) and with immensely greater computational resources.
- The proposed study presents the worst-case analysis. The evaluation of predicates ρ'_o, ρ'_s and ρ'_{rt} within each obligation ξ are evaluated for every $ac_{j-u_i:o_k} \in P_{ac_{j,u_i}}$. Conversely, in a real scenario not all policies ρ include subjects, objects and relationships predicates, thus reducing the measured temporal workload. Additionally, the algorithm applied in the evaluation can be enhanced (recall Alg. 15), e.g. a divide and conquer algorithm to search in ordered lists may be used.
- Computational resources currently applied by WBSNs are much more powerful than those applied herein, e.g. parallelism could alleviate the problem.

In sum, this worst-case analysis has shown that even with constrained computational resources and with heavyweight policies, the proposed approach is feasible.

6 Related work

Lots of WBSN models have been developed. Many of them are based on assorted features, i.e. roles [18], trust [5], relationships [13], attributes [21] and ontology [20]. Besides, dealing with attributes management but looking for expressiveness $SoNeUCON_{ABC}$ was proposed [15].

Other proposals manage data provenance in WBSNs. The origin and traces of data is involved in the access control management process. A data provenance based access control model is proposed by Park et al. [23][24]. It provides dynamic separation of duties, origin-based control and objects versioning in environments like WBSNs. Pei et al. [25] define a framework to capture data provenance and create access control policies from collected data being possible its application in WBSNs. Cheng et al. [9] look for the administrative management of a relationship-based access control model including provenance management.

A step forward can be taken by managing user provenance access control. Considering this feature as a trustworthiness analysis focused on tracing WBSN users' actions, some works can be pointed out. A monitoring system to capture and analyse WBSN users behaviour is proposed in [17]. Sybildefender [32], SybilInfer [11] and Sybilguard [33] focus on identifying sybil WBSN nodes. In addition, [26] works with policies that involve users' actions but they are applied for dynamic access control instead of provenance management.

The negative side of provenance management is the privacy problems it involves [12] as the identification of data or user traces may reveal private data. Multiple proposals work on the establishment of anonymous interactions [31][35]. Others focus on protecting users' data [16][10][2] or users' relationships [34][6] by applying cryptography. Several works in the context of social translucency have been proposed [14]. The idea is the management of which relationships are established and to whom by being aware of the situation and accountable at the same time.

Despite existing WBSN access control models, user provenance has not already been addressed by any of them. In the same way, those which have worked with user provenance in the form of capturing users' behaviours, do not consider privacy at all, in contrast to the concept of translucency proposed in this model.

7 Conclusion

The massive expansion of WBSNs together with the amount of security issues they involve, foster their research and innovation. The origin and trace of actions performed by a WBSN user, called user provenance, together with translucency to avoid privacy problems are requirements to include within WBSN access control models. $SoNeUCON_{ABC}Pro$ extends a previous version, $SoNeUCON_{ABC}$ an expressive access control model for WBSNs, including the management of user provenance together with translucency. From the authors knowledge this is the first time both concepts are applied for access control management purposes. Its implementation has been empirically studied and it is feasible in different scenarios. While translucency management could be performed without restrictions, some settings are acceptable for user provenance management.

Future work will focus on facilitating selective translucency. Users have to be able to choose to whom translucency policies are applied instead of hiding performed actions for everyone. Also the improvement of performance is an issue to consider, as well as usability issues regarding the specification and management of policies by WBSN users.

Acknowledgments

This work was supported by the MINECO grants TIN2013-46469-R (SPINY: Security and Privacy in the Internet of You) and TIN2016-79095-C2-2-R (SMOG-DEV); by the CAM grant S2013/ICE-3095 (CIBERDINE: Cybersecurity, Data, and Risks); and by the Programa de Ayudas para la Movilidad of Carlos III University of Madrid, Spain (J. M. de Fuentes and L. Gonzalez-Manzano grants). Finally, authors would like to thank Security and Privacy Research Group of University of Birmingham for their comments in an early version of this paper.

References

1. The advanced distributed learning (ADL) initiative. Experience api. version 1.0.1. `http://www.adlnet.org/wp-content/uploads/2013/10/xAPI_v1.0.1-2013-10-01.pdf,lastaccessJuly2016`, 2013.
2. Filipe Beato, Markulf Kohlweiss, and Karel Wouters. Scramble! your social network data. In *Privacy Enhancing Technologies*, pages 211–225. Springer, 2011.
3. Peter Buneman, Sanjeev Khanna, and Wang-Chiew Tan. Data provenance: Some basic issues. In *FST TCS*, pages 87–93. Springer, 2000.

4. B. Carminati and E Ferrari. Access control and privacy in web-based social networks. In *International Journal of Web Information Systems*, number 4, pages 395–415, 2008.
5. Barbara Carminati, Elena Ferrari, and Andrea Perego. Rule-based access control for social networks. In *OTM Workshops*, pages 1734–1744. Springer, 2006.
6. Barbara Carminati, Elena Ferrari, and Andrea Perego. Private relationships in social networks. In *ICDE*, pages 163–171. IEEE, 2007.
7. Barbara Carminati, Elena Ferrari, and Andrea Perego. Enforcing access control in web-based social networks. *TISSEC*, 13(1):6, 2009.
8. Y. Cheng, J. Park, and R Sandhu. Relationship-Based Access Control for Online Social Networks: Beyond User-to-User Relationships. In *SocialCom*, pages 646–655, 2012.
9. Yuan Cheng, Khalid Bijon, and Ravi Sandhu. Extended rebac administrative models with cascading revocation and provenance support. In *SACMAT*, pages 161–170. ACM, 2016.
10. Leucio Antonio Cutillo, Refik Molva, and Thorsten Strufe. Safebook: A privacy-preserving online social network leveraging on real-life trust. *Communications Magazine, IEEE*, 47(12):94–101, 2009.
11. George Danezis and Prateek Mittal. Sybilinfer: Detecting sybil nodes using social networks. In *NDSS*, 2009.
12. Susan B Davidson et al. On provenance and privacy. In *EDBT/ICDT*, pages 3–10. ACM, 2011.
13. Philip WL Fong and Ida Siahaan. Relationship-based access control policies and their policy languages. In *SACMAT*, pages 51–60. ACM, 2011.
14. Eric Gilbert. Designing social translucence over social networks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2731–2740. ACM, 2012.
15. Lorena González-Manzano, Ana I González-Tablas, José M de Fuentes, and Arturo Ribagorda. $soneucon_{ABC}$, an expressive usage control model for web-based social networks. *computers & security*, 43:159–187, 2014.
16. Sonia Jahid et al. Decent: A decentralized architecture for enforcing privacy in online social networks. In *PERCOM Workshops*, pages 326–332. IEEE, 2012.
17. Efthymios Lalas, Anastasios Papathanasiou, and Costas Lambrinoudakis. Privacy and traceability in social networking sites. In *PCI*, pages 127–132. IEEE, 2012.
18. J. Li et al. Role Based Access Control for social network sites. In *JCPC*, pages 389–394. IEEE, 2009.
19. S. Lynch. The Agency "Cannot Survive Without Being More Transparent". `https://www.gsb.stanford.edu/insights/former-nsa-head-michael-hayden-agency-cannot-survive-without-being-more-transparent`, `lastaccessJuly2016`, 2014.
20. A. Masoumzadeh and J. Joshi. OSNAC: An Ontology-based Access Control Model for Social Networking Systems. In *SOCIALCOM*, pages 751–759. IEEE Computer Society, 2010.
21. CVD. Munckhof. Content Based Access Control in Social Network Sites. Master's thesis, Eindhoven University of Technology, 2011.
22. Fiona Fui-Hoon Nah. A study on tolerable waiting time: how long are web users willing to wait? *Behaviour & Information Technology*, 23(3):153–163, 2004.
23. Jaehong Park, Dang Nguyen, and Ravi Sandhu. On data provenance in group-centric secure collaboration. In *CollaborateCom*, pages 221–230. IEEE, 2011.
24. Jaehong Park, Dang Nguyen, and Ravi Sandhu. A provenance-based access control model. In *PST*, pages 137–144. IEEE, 2012.

25. Jisheng Pei and Xiaojun Ye. Towards policy retrieval for provenance based access control model. In *TrustCom*, pages 769–776. IEEE, 2014.
26. D. J. Power, M. A. Slaymaker, and A. C. Simpson. Conformance checking of dynamic access control policies. In *ICFEM*, volume 6255 of *Lecture Notes in Computer Science*, pages 227–242. Springer, 2011.
27. Ravi S Sandhu and Pierangela Samarati. Access control: principle and practice. *Communications Magazine, IEEE*, 32(9):40–48, 1994.
28. Roger S Scowen. Extended bnf-a generic base standard. Technical report, Technical report, ISO/IEC 14977. http://www. cl. cam. ac. uk/mgk25/iso-14977. pdf, 1998.
29. R Simcox. Surveillance after snowden: Effective espionage in an age of transparency, 2015.
30. ISO standards. Date and time format - iso 8601, 1988.
31. Latanya Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.
32. Wei Wei et al. Sybildefender: Defend against sybil attacks in large social networks. In *INFOCOM*, pages 1951–1959. IEEE, 2012.
33. Haifeng Yu et al. Sybilguard: defending against sybil attacks via social networks. In *ACM SIGCOMM Computer Communication Review*, volume 36, pages 267–278. ACM, 2006.
34. Yao Zheng et al. Privacy-preserving link prediction in decentralized online social networks. In *ESORICS*, pages 61–80. Springer, 2015.
35. Bin Zhou and Jian Pei. Preserving privacy in social networks against neighborhood attacks. In *ICDE*, pages 506–515. IEEE, 2008.

Appendix: Temporal workload enforcement in $SoNeUCON_{ABC}$

Coloured in gray in Table 6, the evaluation of proposed policies in $SoNeUCON_{ABC}$ takes 13 ms at minimum and 184.5 ms on average.

Table 6. Policy enforcement temporal workload in $SoNeUCON_{ABC}$

WBSN id = 1						
rt id	P1-TW (ms)	P2-TW (ms)	P3-TW (ms)	P4-TW (ms)	P5-TW (ms)	P6-TW (ms)
1	4143	4144	4143	4143	4142	4142
2	435	435	435	435	435	435
3	28	28	28	28	28	28
4	54	55	54	54	54	54
5	38	38	38	38	38	38
6	51	51	51	51	51	51
7	13	13	13	13	13	13
WBSN id = 2						
rt id	P1-TW (ms)	P2-TW (ms)	P3-TW (ms)	P4-TW (ms)	P5-TW (ms)	P6-TW (ms)
9	21291	21304	21291	2289	21289	21287
9	712	712	712	713	712	712
10	58	57	57	57	58	57
11	88	88	89	89	88	88
12	61	60	60	60	61	60
13	62	62	63	62	62	62
14	31	30	30	30	30	30
WBSN id = 3						
rt id	P1-TW (ms)	P2-TW (ms)	P3-TW (ms)	P4-TW (ms)	P5-TW (ms)	P6-TW (ms)
15	56825	56816	56815	56813	56820	56811
16	274	273	274	273	273	273
17	80	80	80	81	80	80
18	110	111	110	110	110	110
19	89	88	89	88	88	88
20	86	86	86	86	87	86
21	36	37	36	37	36	36
WBSN id = 4						
rt id	P1-TW (ms)	P2-TW (ms)	P4-TW (ms)	P5-TW (ms)	P6-TW (ms)	P7-TW (ms)
22	105549	105545	105554	105558	105496	105563
23	1721	1722	1721	1721	1721	1722
24	44	45	44	44	44	45
25	135	134	134	134	134	135
26	96	97	96	96	96	97
27	83	83	83	83	83	84
28	46	46	46	46	46	47
Average	184.6	184.5	184.5	184.5	184.5	184.6
Total average	184.5					

*Results of evaluating proposed policies (see [15]) in created WBSNs over 28 pairs of random users are presented in this Table. Considering 2000 ms a usability limit for being approximately the tolerable waiting time of WBSN users for information retrieval [22], removing cases that exceed this threshold (details in [15]).