Copyright © 2019 Tech Science Press CMC, vol.59, no.2, pp.533-546, 2019

Personalized Privacy Protecting Model in Mobile Social Network

Pingshui Wang[1, *], Zecheng Wang[1], Tao Chen[1, 2] and Qinjuan Ma[1]

Abstract: With the rapid development of the new generation of information technology, the analysis of mobile social network big data is getting deeper and deeper. At the same time, the risk of privacy disclosure in social network is also very obvious. In this paper, we summarize the main access control model in mobile social network, analyze their contribution and point out their disadvantages. On this basis, a practical privacy policy is defined through authorization model supporting personalized privacy preferences. Experiments have been conducted on synthetic data sets. The result shows that the proposed privacy protecting model could improve the security of the mobile social network while keeping high execution efficiency.

Keywords: Mobile social network, privacy policy, personalized privacy preference, models.

1 Introduction

With the rapid development of Internet, cloud computing, big data and artificial intelligence in the new generation of information technology, especially the application of Web 2.0 techniques, mobile social networks (MSNs) have experienced exponential growth in recent years. All kinds of social network products were introduced to the Internet, such as Facebook, Twitter, Myspace, RenRen, Microblog, WeChat, QQ, etc. MSNs provide users with a platform for communication, sharing information, making friends. With the popularity and development of social networks, social networking sites store a large number of users' personal data, which brings much convenience to data analysis. At the same time, it also causes great threat and challenge to individuals' privacy, because MSNs data may contain personal private information. Protecting the privacy of users against unwanted disclosure in such circumstance poses challenging problems. Issues on privacy disclosure are the greatest threat to the personal information security in the era of big data [Garcia, Goel, Agrawal et al. (2018); Heravi, Mubarak and Raymond (2018); Liu, Wang and Yang (2014); Rathore, Sharma and Loia (2017); Wang, Sun and Ma (2012); Yang, Huang, Li et al. (2018)].

In recent years, the issues on privacy protection in mobile social network are deeply researched, and lots of effective privacy preserving technologies have been developed.

[1] Anhui University of Finance and Economics, Bengbu, 233030, China.

[2] University of Kansas, Lawrence, Kansas, 66045, USA.

[*] Corresponding Author: Pingshui Wang. Email: 120081049@aufe.edu.cn.

CMC. doi:10.32604/cmc.2019.05570 www.techscience.com/cmc

The existing researches on mobile social network privacy protection concentrate mainly on privacy preserving data publishing, data mining and access control [Cheng, Park and Shu (2016); Kokciyan and Yolum (2016); Kumar and Kumar (2017); Schlegel, Chow, Huang et al. (2017); Soliman, Bahri and Girdzijauskas (2016); Sun, Yu, Kong et al. (2014); Such and Criado (2016); Tai, Yu, Yang et al. (2011); Thapa, Liao, Li et al. (2016); Wang, Srivatsa and Liu (2012); Zou, Chen and Ozsu (2009)], in which anonymization is the main privacy preserving technology for social network data release, so that the data released can meet the need of data analysis while user privacy is not compromised; and social network access control techniques mainly focuse on designing social network access control model to solve the problem of social network data access authorization[Adam, Atluri, Bertino et al. (2002); Carminati, Ferrari and Perego (2006); Cirio, Cruz and Tamassia (2007); Jayaraman, Rinard and Tripunitara (2011); Li, Tang and Mao (2009); Ma, Tao, Zhong et al. (2016) ; Yuan and Tong (2005)]. However, there is relatively less research work on personalized privacy protection of social network data, so that it increases the risk of privacy disclosure and the complexity of user privacy settings. In this paper, we summarize the main access control models in mobile social network, analyze their contribution and point out their disadvantages. On this basis, a practical privacy policy is defined through authorization model supporting personalized privacy preferences.

The rest of the paper is organized as follows: In Section 2, we analyze the main access control models in mobile social network; In Section 3, we introduce some concepts about privacy policy definition; Section 4 provides a personalized privacy policy description and authorization model; In Section 5, we conduct a privacy policy conflict analysis; In Section 6, we design a personalized privacy policy management system for social network; Section 7 contains our conclusions and future work.

2 The related work of access control model in mobile social network

Access control in mobile social network is one of the most common manners of users' privacy protection. Several access control models have been proposed. We review them briefly as follows.

Role based access control model [Li, Tang and Mao (2009)] implements access control according to a pre-set role and the corresponding access privilege. However, the method mainly aims at the determined user community and cannot solve the problem of access authorization to unknown users and dynamic resources.

Attribute based access control model can provide a better solution to the above problem [Adam, Atluri and Bertino (2001); Cirio, Cruz and Tamassia (2007); Yuan and Tong (2005)]. It realizes the dynamic access control in open environment using a set of attribute authorization rules based on the subject attribute, object attribute and environment attribute constraints. But the model is only applied to the situation that the owner and manager of resource are integrated in the same social network, in which access control policy is developed by the manager of resource, so it is not suitable for the condition that the owner and manager of resource are separated, and it cannot satisfy the requirement of social network users' personalized privacy preferences.

Rule based access control model [Carminati, Ferrari and Perego (2006)] defines the relationship between the visitor and owner of resource, the maximum topological distance and minimum confidence and other restrictions by rules, so that the automatic and flexible access control is achieved on the basis of rules reasoning. But due to the large number of rules, it is prone to result in conflicted policy and cannot guarantee the consistency of authorization and the effective implementation of policies.

Authorization rules based access control model [Jayaraman, Rinard and Tripunitara (2011); Ma, Tao, Zhong et al. (2016)] adds the concepts of user attributes and permissions allocation rules on the basis of rule based access control model. It achieves the dynamic role permission assignment, but the model does not meet the demand of user-defined privacy policies.

In view of the above problems, we have proposed an authorization model for personalized privacy preferences, which describes user privacy preference by using first-order logic and supports user-defined personalized dynamic privacy policy, strategy analysis of consistency of automation by using logic programming method and the implementation of authorization inference rules.

3 Basic concepts about privacy policy definition

Privacy policies are mainly composed of the subject, object, action and the restriction conditions that the access authorization needs to meet, which are introduced briefly in this following.

Definition 1 (Subject). Subject refers to the visitors who access the social network resources, and all subject set is represented as *UserS*. The feature of the subject is called attribute, which consists of attribute name and attribute value. For example, attributes of the subject include identity card number, name, gender, birthday, phone, email, address, education, hobbies, etc..

Definition 2 (Object). Object refers to the secret resources that visitors try to access, and all object set is represented as *ResS*. The secret resources are divided into attribute information and data resources according to the type of resources, and data resources refers to the user's publishing information such as personal diary, speech, comment, photo, video, etc..

Object tag is composed of tag name and tag value, which is the identification method of data resources. Because the user has many data resources, to realize flexible grouping and segmentation of data resources, data resources for users can add labels such as “type”, “time”, “place” , “importance”, and so on.

Definition 3 (Action). Action refers to the subject’s operation executed on the object, such as accessing, reading, commenting, sharing, etc., and all action set is represented as *ActS*.

Definition 4 (Permission). Permission refers to the action performed on an object, which is denoted as $<r, a>$, in which $r \in ResS$ and $a \in ActS$. All permission set is expressed as *PerS*.

Definition 5 (Role). Role is the user group marked according to the demand for subject’s attribute, and all role set is represented as *RoleS*. The directly logical relationship between subject and permission can be insulated by correlating the role and authority.

Definition 6 (Role Hierarchy, RH). A partial order relation defined on the set of roles. We assume that there are n groups of roles, such as ($role_1,\cdots,role_n$), RH: $role_i \times role_j$, in which $role_i$ and $role_j$ belongs to *RoleS*, $role_i$ is called a senior role and $role_j$ is lower when $role_i$ is greater than or equal to $role_j$, the senior role inherits permissions from the lower, and the lower role succeeds users from the senior

Definition 7 (Predicate). Description that an entity has a property or a relationship exists among multiple entities, which consists of two parts: the name of predicate and the parameters, expressed as *Predicate* (x_1, x_2,.,x_n), in which x_i(i=1,., n) may be constant, variable or first-order predicate, the set of all the predicate is denoted as *PredS*.

For example: *Is* (*x.role, 'teacher'*), which means that the role of the subject *x* is a teacher; *Is* (*y.tag, 'red'*), which means that the tag of the object *y* is *red*.

Definition 8 (Constraint). Basic conditions and limitations that need to be met for access permission, which may be predicate or predicate logic expression, expressed as:$prd_1\Theta$ $prd_2\cdots\Theta prd_n$, Θ may be "*and*" or "*or*" logical operators, prd_i belongs to *PredS*(i=1,., n).

According to the constraint content, the constraints are divided into 3 categories: subject attribute constraints, object tag constraints and environmental constraints.

- Subject attribute constraints refer to the subject's age, gender, address, profession, hobby and other constraints, and all the subject attribute constraint set is expressed as *SAttrC*. For example, *Larger* (*x.age*, '18') ^*Is* (*x.gender, 'male'*), requires subject "*x*" be male over 18 years old;
- Object tag constraints refer to the object tag condition required for authorization of data resources, and all the object tag constraints set is represented as *RtagC*. For example, *Is* (*y.type, 'video'*) ^*Is* (*y.tag, 'workshop'*), means the object access authorization is limited to workshop video;
- Environmental constraints refer to constraints such as time, address, system state, context, etc., and all the environment constraints set is represented as *EnvC*. For example, *TimeWithin* ('14:00', '18:00') represents the time constraint [14:00, 18:00]; participated(*x,'workshop'*) indicates that the subject *x* participated in workshop. Introducing environmental constraints for the development of privacy policy may provide the access control strategy with real-time and good interaction, and improve the security of user private resources.

4 Authorization model supporting personalized privacy preferences

In this section, we propose an authorized model supporting personalized privacy preferences, which extends role-based access control models and adds subject attributes based visitor role authorization rules and role permissions assignment rules based on object tags, as shown in Fig. 1. In the model, according to authorization rules of the visitor role, visitors meeting the subject attribute constraints obtain permissions; according to role permission authorization rules, visitors meeting the object label constraints are assigned to the corresponding role, and they also contain inherited permissions caused by the role hierarchy relationship. So visitors obtain permissions by user role and hierarchy. The relevant rules are defined as follows.

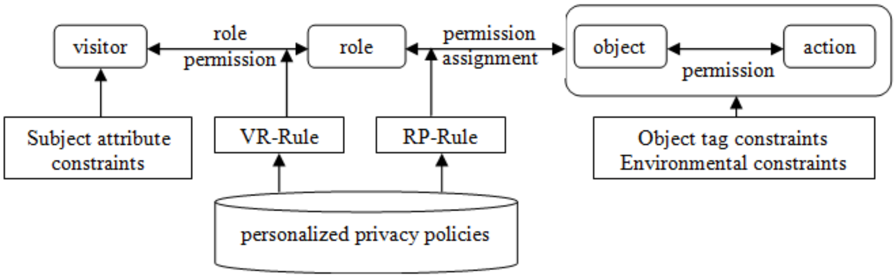


Figure 1: An authorized model supporting personalized privacy preferences

Definition 9 (Visitor role permissions rules, VR-Rule). *assign_role* (*u, role*) ← Q_1x_1 ··· Q_mx_m ($sc_1\Theta sc_2\cdots\Theta sc_n$), in which *u*∈*UserS*, *role*∈*RoleS*, Θ may be logical operators such as “and”(^) or “or”(v), sc_i ∈*SAttrC*, *i*=1, ···, *n*. x_j is an instance variable, *j*=1, ···, *m*, Q_i ∈ {∃,∀}; ∃ is existential quantifier and ∀ is universal quantifier, that is the visitor “*u*” gets the role “*role*” while satisfying all subject attribute constraints.

Example 1. *assign_role* (*x,'college classmate'*) ←∀*x* *Larger*(*x.age,*'18') ^*Is* (*x.class,*'12*computer*-1') ^ *Is* (*x.graducate,'AUFE'*) means that peoples satisfy the rule of over 18 years old and come from ‘*AUFE’* and '12*computer*-1' are ‘*college classmate’*.

Definition 10 (Role permission assignment rules, RP-Rule). *P_assign*[*D_assign*] (*role, r, a*) Q_1x_1 ··· Q_mx_m ($re_1\Theta re_2\cdots\Theta re_n$), in which *role*∈*RoleS*, *r*∈*ResS*, *a*∈*ActS*, Θ may be logical operators such as“and”(^) or “or”(v), re_i∈{*RtagC*; EC}, *i*=1, ···, *n*. x_j is an instance variable, *j*=1, ···, m, Qi ∈{∃,∀}; ∃ is existential quantifier and ∀ universal quantifier, *P_assign* positive and *D_assign* negative authorization, that is in an *EC* environment, meeting all constraints object permissions for the <*r*, *a*> is/[is not] assigned to role “*role*”.

Examples 2. *P_assign*('*college classmate*', *y,'comment'*) ←∃*y* *Is*(*y.type,'photo'*) ^ *Is*(*y.tag,'graduation'*), said the college classmates can comment graduation photos.

Definition 11 (privacy policy). The same user defined the set of visitor-role authorization rules and role- permission assignment rules.

Example 3. privacy policy ={*VR-Rule1*, *PR-Rule1*}, *VR-Rule1*: *assign_role* (*x,'college classmate'*) ←∀*x* *Is* (*x.class,'12computer-1'*) ^ *Is* (*x.graducate,'AUFE'*); *PR-Rule1*: *P_assign*('*college classmate*', *y,'comment'*) ←∃*y* *Is*(*y.type,'photo'*) ^ *Is*(*y.tag,'graduation'*). The information of user Alice is stored in the system as follows: *name*=’*Alice’*, *class* =’12*computer*-1’, *graducate* =’*AUFE’*; the label of object photo1 is described as follows: *type*= ‘*photo’*, *tag* ={*graduation, AUFE, 12computer-1*}. *Alice* is assigned the role of college classmate according to *VR-Rule1*, at the same time, if resources *photo1* meet the object tag constraint, *Alice* gets permission to comment on photo1 through the role of college classmate.

The advantage of authorized model supporting personalized privacy preferences is reflected in two aspects: on the one hand, the privacy policy is descripted on the basis of first-order logic, which could meet users’ personalized privacy needs, such as the fine-grained authorization requirements of resources protection, clearly express the clear semantic demand of user's privacy will and support the strategy reasoning authorization without an explicit description; on the other hand, access control is achieved for the unknown users in social networks and a large number of dynamic data resource. The reasoning based on the role authorization of subject attribute constraints can realize the automatic and dynamic visitor role authorization, which could solve the problem of access request of unknown users in social network; meanwhile, users in social networks have many data resources, which are often added and modified, so the traditional authorization model aiming at the specific resources becomes difficult to maintain. The proposed role permission assignment rules based on object tag constraints can realize the permissions assignment to large, dynamic resources.

However, due to the overlapping or hierarchical relationship among rules' subject attributes, resource attributes and action attributes, there may be logical inconsistencies in the formulation of privacy policies, for example, both positive authorization and negative authorization maybe exist on the same subject and object in different strategies, which will result in the privacy policy conflict.

5 Privacy policy conflict analysis

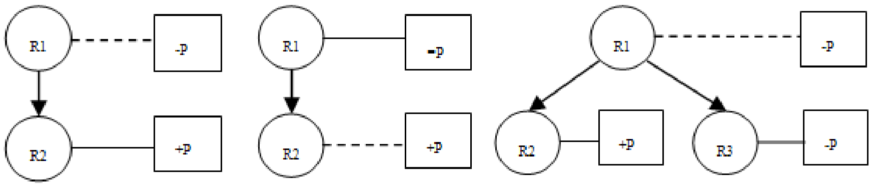
Due to the overlap or hierarchical relationship among the subject attributes, resource attributes and action attributes of rules, there may be logic inconsistencies in the formulation of privacy policies. For example, in different policies, there are both positive authorization and negative authorization for the same subject and object, which results in privacy policy conflicts. According to whether the cause of policy conflict is related to specific data, it can be divided into logical conflict and instance conflict.

5.1 Logical conflict

Logical conflict refers to the logical inconsistency in the process of policy definition, such as role contradiction delegation, which refers to the logical conflict that the same role is assigned both positive and negative authorization.

Example 4. privacy policy={*VR-Rule1, PR-Rule1, PR-Rule2*}, where *VR-Rule1*:*assign_role*(*x,'groupmember'*)←∀*x* *Is*(*x.project,'mobile Application'*)； *PR-Rule1*:*P_assign* (*'groupmember',y, 'read'*) ← ∃*y* *Is*(*y.type,'log'*) ^ *Is*(*y.tag, 'work'*) ^ *TimeWithin*('8:00AM', '6:00PM')； *PR-Rule2*: *D_assign*(*'groupmember',y, 'read'*)←∃*y* *Is*(*y.type,'log')* ^ *Is*(*y.tag, 'work'*) ^ *DayWithin*(*'Saturday', 'Sunday'*). *VR-Rule1* indicates that groupmembers are participated in the same project; *PR-Rule1* indicates that groupmembers can view the work log from 8:00 to 18:00; *PR-Rule2* indicates that groupmembers cannot view the work log on weekends.

Another typical logical conflict is the privilege inheritance conflict, which is the contradiction between authorization and explicit authorization caused by role hierarchy. As shown in Fig. 2, the circle represents the role, the square represents the privilege, +P and-P represent the positive and negative authorization to the same resource respectively, the arrow represents the role hierarchy, and the solid line indicates an existing role-permission assignment relationship, the dashed line represents the newly added role-privilege assignment relationship. According to the inheritance relationship of permissions in the role hierarchy [Wang, Sun and Ma (2012)], when lower-level roles are assigned positive authorization, higher-level roles inherit positive authorization from lower-level roles to higher-level roles according to positive authorization. If negative authorization is added to higher-level roles, it would conflict with the positive authorization of lower-level roles and cause policy conflicts, such as Fig. 2(a) . When lower-level roles are assigned negative authorization, the added positive authorization of higher-level roles does not cause policy conflicts. When high-level roles are assigned negative authorization, the negative authorization of high-level roles to resources must imply negative authorization of low-level roles according to negative authorization propagation from high-level to low-level, and if the positive authorization of low-level roles is added, it would conflict with the positive authorization of high-level roles, resulting in policy conflicts, as shown in Fig. 2(b); When high-level roles include multiple low-level roles and there are mutually exclusive privileges between low-level roles, if a new negative authorization is added to high-level roles, it would conflict with the negative authorization of low-level roles and cause policy conflicts, as shown in Fig. 2(c).



(a) (b) (c)

Figure 2: Example of permission inheritance conflict

Example 5. privacy policy ={*VR-Rule1*, *VR-Rule1*, *VR-Rule2*, *PR-Rule1*, *PR-Rule2*}, where *VR-Rule1*: *assign_role*(*x*,'*schoolmate*')←∀*x* *Is*(*x.graducate*,'*AUFE*')； *VR-Rule2*: *assign_role*(*x*,'*classmate*')←∀*x* *Is*(*x.graducate*, '*AUFE*') ^ *Is*(*x.class*,'*12computer-1*'); *PR-Rule1*: *P_assign*('*schoolmate*',*y*,'*tag*')←∃*y* *Is*(*y.type*, '*log*') ^ *Is*(*y.tag*, '*personal*'); *D_assign*('*classmate*',*y*, '*tag*')←∃*y* *Is*(*y.type*, '*log*') ^ *Is*(*y.tag*,'*personal*'). *VR-Rule1* means schoolfellows who graduated from *AUFE*; *VR-Rule2* means students who graduated from *AUFE* and whose class name is 12*computer*-1; *PR-Rule1* means schoolfellows can mark personal logs; and *PR-Rule2* students can't mark personal logs. According to the rule of *VR-Rule 1* and *VR-Rule 2*, classmates are greater than or equal to schoolfellows, due to the role hierarchy, the authorization of classmates to inherit the role of schoolfellows can mark personal logs, but the explicit definition of *PR-Rule 2* indicate that classmates cannot mark personal logs, which results in policy conflicts.

5.2 Instance conflict

Instance conflict means that there is no logical conflict on the policy definition itself, but there are policy conflicts caused by the instances in the database which trigger policy conflict conditions. In the authorization model that supports personalized privacy preference, users are authorized by VR-Rule and PR-Rule. In the process of defining two kinds of rules, there may be a user instance that satisfies two kinds of role constraints simultaneously, which leads to the application of two opposite strategies at the same time and results in policy conflicts.

Example 6 privacy policy ={*VR-Rule1*,*VR-Rule2*, *PR-Rule1*, *PR-Rule2*}, where *VR-Rule1*: *assign_role* (*x*,'*college classmate*') ←*Is* (*x.class*,'*12computer-1*') ^ *Is* (*x.graducate*,'*AUFE*');

VR-Rule2: *assign_role*(*x*,' *groupmember*')←∀*x* *Is*(*x.project*, '*mobile Application*')； *PR-Rule1*: *P_assign*('*college classmate*', *y*,'*comment*') ←∃*y* *Is*(*y.type*,'*photo*') ^ *Is*(*y.tag*,'*graduation*'); *PR-Rule2*: *D_assign*('*groupmember*', *y*, '*read*') ← ∃*y* *Is*(*y.type*,'*photo*') ^ *Is*(*y.tag*, '*red*'). *VR-Rule1* indicates that people in the class of 12*computer*-1 and graduate from *AUFE* are college classmates; *VR-Rule2* indicates that people involved in the same project are groupmembers; *PR-Rule1* indicates that college classmates can comment on graduation photos; and *PR-Rule2* indicates that groupmembers cannot view photos marked red. The information of user *Alice* is stored in the system as follows: *name*=’*Alice’*, *class* =’*12computer*-1’, *graducate* =’*AUFE’*, *project* =’*mobile Application’*, *Alice* satisfies the subject constraints of two roles: ‘*college classmate’* and ‘*groupmember’*, and *Alice* has two roles at the same time; the label of object photo1 is *type* = ‘*photo’*, *tag* = {*graduation*, *red*}, According to *PR-Rule 1*, *Alice* can comment on photo *photo1*, but *PR-Rule 2* makes it impossible for *Alice* to view photo *photo1*, which causes a policy conflict.

6 Design of personalized privacy policy management system for social network

In order to effectively integrate the authorization model supporting personalized privacy preference into the existing social network system, we design a personalized privacy policy management system, which allows users to define personalized privacy policies and implement access control based on privacy policy. The system structure diagram is shown in Fig. 3. The main components include subject attribute base, resource attribute base, environment attribute base, privacy policy base, policy management module, and policy evaluation module, etc., among which the subject attribute base is used to storage user's principal attributes, resource attribute base is used to storage resource attributes and resource labels, environment attribute base is used to storage context environment information, and privacy policy base is used to storage privacy policies.

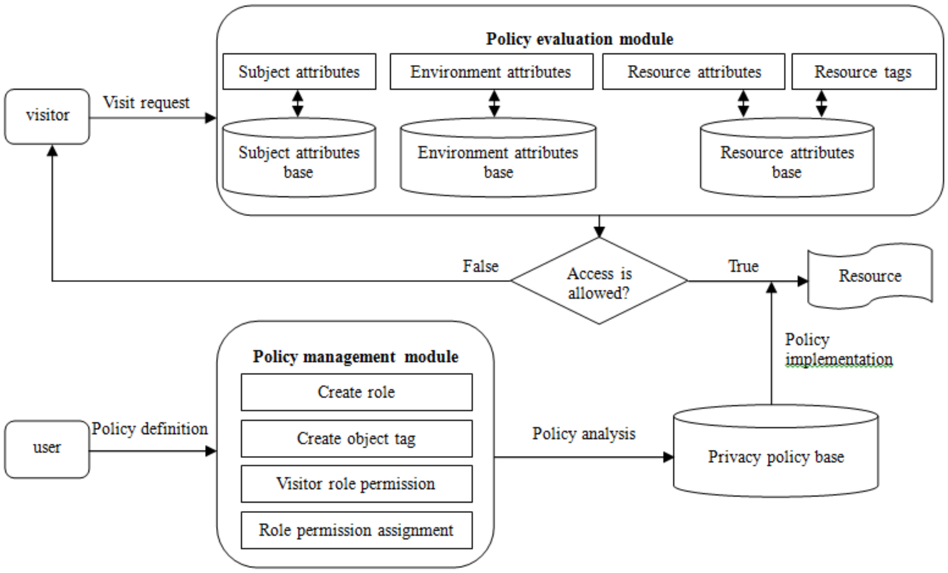


Figure 3: The system structure diagram

6.1 Policy management module

Policy management module implements personalized privacy policy definition. The main functions include: create new roles and role hierarchies by creating roles; add data resource tags by creating resource tags; define the range of role subject constraints for role assignment through user-role authorization; define the object constraints, authorization action and authorization role to complete the permission assignment by role-permission assignment; and provide the functions of modification and deletion of the above definition.

6.2 Policy analysis module

Policy analysis module implements automatic strategy conflict detection. The main functions include: rule parsing, which automatically parses user privacy policy and database data into Prolog facts and stores the results in Prolog files. In the part of rule query analysis, users make query analysis requests according to the predefined policy conflict rules, judge whether the policy meets the consistency according to the query results, and correct the conflicting policy prompts to ensure the correct implementation of the policy.

6.3 Policy evaluation module

Policy evaluation module implements policy-based access control. When a visitor sends a request to access a resource, it inquires the information of the subject property, the resource attribute and the environment property, and matches the visitor's principal attribute with the subject attribute constraint to get the visitor role, traverses the authorization permission of the role set, and extracts the information about the access resources in the database, matches object attributes and tags in the permission set. If the match is successful, the resource is open to the visitor, and the specified operation is performed, otherwise the visitor's request is rejected and the results of the access evaluation are fed back to the visitor.

6.4 The system database

The system database is mainly composed of subject table (*subject*), role table (*role*), object table (*object*), object tag table (*object_tag*), action table (*action*), permission table (*permission*), subject-role assignment table (*S-R*) and role-permission assignment table (*R-P*). Through user role assignment table, the relationship between user table and role table is mapped to “one to one”, “one to many” or “many to many”. In the same way, role-permission assignment table also maps the relationships of “one to one”, “one to many” or “many to many” between role table and permission table, and the database E-R diagram is shown in Fig. 4.

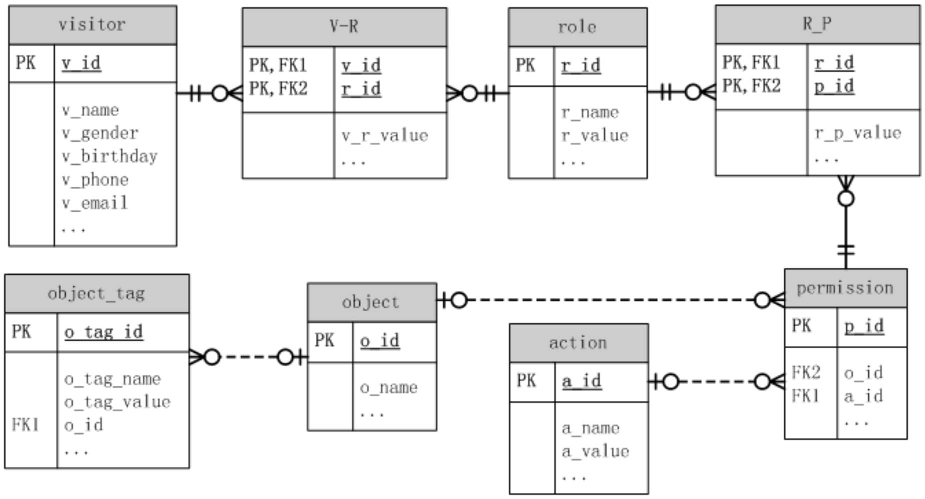


Figure 4: The database E-R diagram

6.5 The system implementation

In order to facilitate the implementation of policy definition for non-professional users through a visual interface, we have developed a personalized privacy policy management system for social networks. Experiments have been conducted on synthetic data sets. The result shows that the proposed privacy protecting model could improve the security of the mobile social network while keeping high execution efficiency. The system experimental environment is described as follows. CPU: Intel(R) Core(TM) i7-6500U @2.50GHz, RAM: 8G, software environment: Windows 7, development language: Python 3.64, Database System: SQL-Server 2012.

Aiming at different ways of conflict query, we first test the impact of the number of users on query performance. Suppose that the user information table has 10 attributes, according to each additional 10 users for a group of experiments, each group of queries carry on 50 tests, we calculate the average query time of 10 rounds. The experimental results are shown in Fig. 5, where the direct conflict query refers to querying instance conflict rules directly without setting query restriction range, that is *conflict* (*User, Object, Action*), represented by dotted lines. Personalized query refers to restricting certain variables to query user authorization path rules, that is *conflict* (*User, photo1, Action*), restricting *Object* = *photo1*, which is expressed in a straight line. The experimental results show that with the increase of the number of users, the direct conflict query time increases linearly, because the direct conflict query is detected by enumeration, the number of users increases, and the enumeration query number increases correspondingly, which leads to the rapid growth of the query time; comparing with the direct conflict query execution, the personalized query is more efficient than direct conflict query, because personalized queries limit some variables, which can quickly locate the causes of policy conflicts, and is less affected by the number of users.

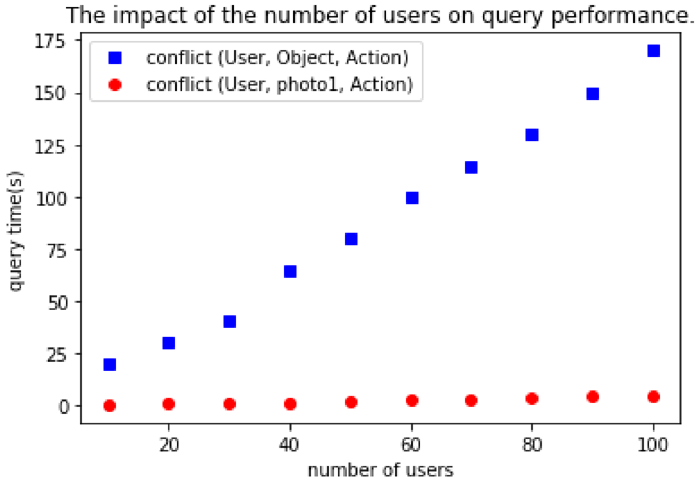


Figure 5: The impact of the number of users on query performance

Secondly, we test the impact of resource quantity on query performance. The number of selected users is 100. The experimental results are shown in Fig. 6, where the direct conflict query is *conflict (User, Object, Action)*, represented by dotted lines, and the personalized query named as *query_per (User, Role, Permission Limit, Object, Action)* is restricted to *User ='Alice'*, represented by a straight line. The experimental results show that with the increase of the number of resources, the direct conflict query time increases linearly first and then gradually stabilizes, because the access authorization of the model is aimed at satisfying all the resources of the object label, not the authorization of a resource. Although the number of resources increases, the policy conflict query time is relatively stable when the resources satisfying the object label constraint are determined; personalized query is more efficient than direct conflict query, and less affected by resource quantity.

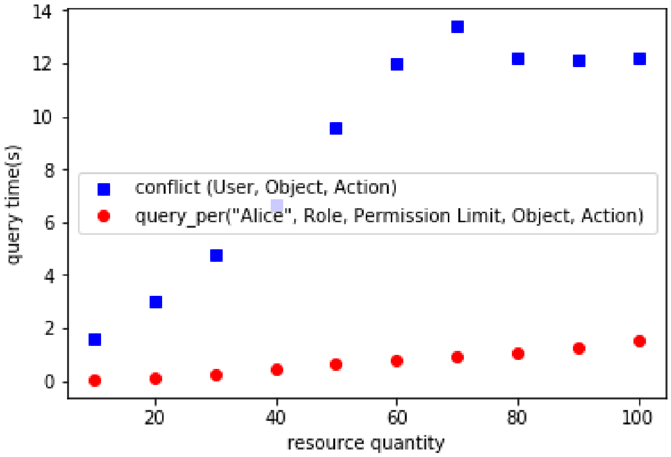


Figure 6: The impact of the resource quantity on query performance

Finally, we test the effect of the number of users on the performance of personalized queries under different conditions. The experimental results are shown in Fig. 7. Queries with three variables, *query_per ('Alice', Role, Permission Limit, 'photo1','read')*, are restricted by three query conditions: *User = 'Alice', Object = 'photo1', Action = 'read'*, whose query time is represented by a solid line. Queries restrict one query condition, that is *query_per (User, Role, Permission Limit, 'photo1', Action)*, which indicates that a query condition *Object*= '*photo1'* is restricted, and its query time is represented by a dotted line. The experimental results show that personalized query has high execution efficiency, and the more restrictive query conditions, the better query performance.

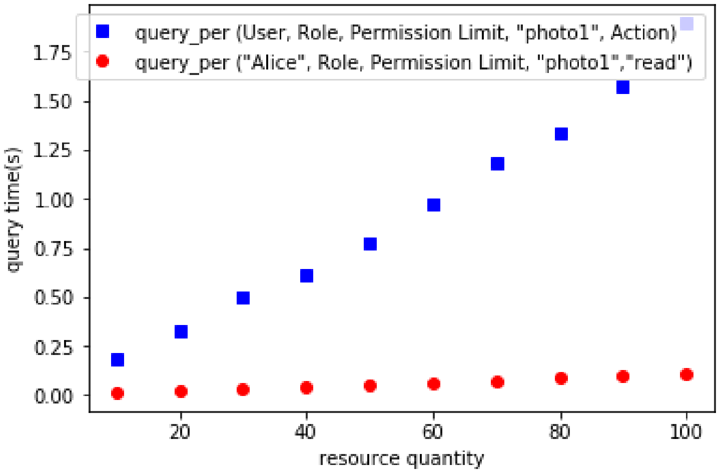


Figure 7: The impact of the resource quantity on query performance of personalized query

7 Conclusions and future work

In recent years, privacy preservation has been widely concerned in academic and industrial fields. Many privacy preservation techniques in mobile social network have been proposed. In this paper, we summarize the main access control models in mobile social network, analyze their contribution and point out their disadvantages, on this basis, a practical privacy policy is defined through authorization model supporting personalized privacy preferences, which can meet the user's personalized privacy policy needs. However, due to the overlapping or hierarchical relationship among rules' subject attributes, resource attributes and action attributes, there may be logical inconsistencies in the formulation of privacy policies, for example, both positive authorization and negative authorization maybe exist on the same subject and object in different strategies, which will result in the privacy policy conflict. The next step is fully analyze the possible conflict between the privacy strategy and comprehensively consider the strategy conflict brought by the resource level relationship, meanwhile, we verify the feasibility of the model by experiments on real datasets .

Acknowledgment: We thank the anonymous reviewers and editors for their very constructive comments. This work was supported by the National Social Science Foundation Project of China under Grant 16BTQ085.

References

Adam, N.; Atluri, V.; Bertino, E.; Ferrari, E. (2002): A content-based authorization model for digital libraries. *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 296-315.

Carminati, B.; Ferrari, E.; Perego, A. (2006): Rule-based access control for social networks. *Lecture Notes in Computer Science, on the Move to Meaningful Internet Systems: OTM'06 Workshops*, vol. 4278, pp. 1734-1744.

Cheng, Y.; Park, J.; Shu, R. (2016): An access control model for mobile social networks using user-to-user relationships, *IEEE Transactions on Dependable and Secure Computing*, vol. 13, no. 4, pp. 424-436.

Cirio, L.; Cruz, I.; Tamassia, R. (2007): A role and attribute based access control system using semantic web technologies. *Proceedings of International Federation for Information Processing Workshop on Semantic Web and Web Semantics*, pp. 1256-1266.

Garcia, D.; Goel, M.; Agrawal, A.; Kumaraguru, P. (2018): Collective aspects of privacy in the Twitter social network. *EPJ Data Science*, vol. 7, no. 1.

Heravi, A.; Mubarak, S.; Raymond, C. (2018): Information privacy in online social networks: uses and gratification perspective. *Computers in Human Behavior*, vol. 84, pp. 441-459.

Jayaraman, K.; Rinard, M.; Tripunitara, M. (2011): Automatic error finding in access-control policies. *Proceedings of 18th ACM Conference on Computer and Communications Security*, pp. 17-21.

Kokciyan, N.; Yolum, P. (2016): PriGuard: a semantic approach to detect privacy violations in online social networks. *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 10, pp. 2724-2737.

Kumar, S.; Kumar, P. (2017): Upper approximation based privacy preserving in online social networks. *Expert Systems with Applications*, vol. 88, pp. 276-289.

Li, J.; Tang, Y.; Mao, C. (2009): Role based access control for social network sites. *Proceedings of Joint Conferences on Pervasive Computing*, pp. 389-394.

Liu, X.; Wang, B.; Yang, X. (2014): Survey on privacy preserving techniques for publishing social network data, *Journal of Software*, vol. 25, no. 3, pp. 576-590.

Ma, L.; Tao, L.; Zhong, Y.; Gai, K. (2016): RuleSN: Research and application of social network access control model. *Proceedings of the IEEE International Conference on Intelligent Data and Security*, pp. 418-423.

Rathore, S.; Sharma, P.; Loia, V. (2017): Social network security: Issues, challenges, threats, and solutions. *Information Sciences*, vol. 421, pp. 43-69.

Schlegel, R.; Chow, C.; Huang, Q, Wong, D. (2017): Privacy-preserving location sharing services for social networks. *IEEE Transactions on Services Computing*, vol. 10, no. 5, pp. 811-825.

Soliman, A.; Bahri, L.; Girdzijauskas, S. (2016): CADIVa: cooperative and adaptive decentralized identity validation model for social networks. *Social Network Analysis and Mining*, vol. 6, no. 1, pp. 1-22.

Such, J. M.; Criado, N. (2016): Resolving multi-party privacy conflicts in social media. *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 7, pp. 1851-1863.

Sun, C.; Yu, P.; Kong, X.; Fu, Y. (2014): Privacy preserving social network publication against mutual friend attacks. *Transactions on Data Privacy*, vol. 7, no. 2, pp. 71-97.

Tai, C.; Yu, P.; Yang, D.; Chen, M. (2011): Privacy-preserving social network publication against friendship attacks. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1262-1270.

Thapa, A.; Liao, W.; Li, M.; Li, P. (2016): SPA: A secure and private auction framework for decentralized online social networks. *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 8, pp. 2394-2407.

Wang, T.; Srivatsa, M.; Liu, L. (2012): Fine-grained access control of personal data. *ACM Symposium on Access Control Models and Technologies*, pp. 145-156.

Wang, Y.; Sun, Y.; Ma, L. (2012): Specification and enforcement of personalized privacy policy for social network. *Journal on Communications*, vol. 33, no. z1, pp. 239-249.

Yang, Z.; Huang, Y. F.; Li, X.; Wang, W. Y. (2018): Efficient secure data provenance scheme in multimedia outsourcing and sharing. *Computers, Materials & Continua*, vol. 56, no. 1, pp. 1-17.

Yuan, E.; Tong, J. (2005): Attributed based access control (ABAC) for web services. *Proceedings of the IEEE International Conference on Web Services*, pp. 561-569.

Zou, L.; Chen, L.; Ozsu, M. (2009): K-automorphism: a general framework for privacy preserving network publication. *Proceedings of the 35th International Conference on Very Large Databases*, pp. 946-957.