Journal of Computing and Security

Research Article, 2023, Volume 10, Number 2 (pp. 1–12)

http://www.jcomsec.org

New Attribute Relation-Based Access Control System via Hybrid Logic

Fatemeh Nazerian [a,*] , Homayun Motameni [b]

[a] *Department of Computer Engineering, Qaemshahr Branch, Islamic Azad University, Qaemshahr, Iran.*
[b] *Department of Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran.*

ARTICLE INFO.

Article history:
Received: 8 February 2023
Revised: 25 June 2023
Accepted: 15 July 2023
Published Online: 2 Oct 2023

Keywords:
Social Network, Hybrid Logic, Policy, Attribute, Breadth-first search, Depth-first search.

ABSTRACT

In recent years, Online Social Network (OSN) has been rapidly evolving and attracted many users. In OSN, users share sensitive information; therefore, effective access control models are needed to protect information from unauthorized users. Currently, Relational Based Access Control (ReBAC) is used to protect user's private information. The authorization policy in ReBAC is based on the relationship type and depth among users; however, it is not sufficient to protect private information such as location, time, and age. In this paper, attributes are added to the social graph to establish an efficient access control in OSN, then a policy model is proposed for the new Attribute Relation Based Access Control model (A-ReBAC), and unambiguous Hybrid Logic (HL) policy language is used to formulate the access control policy model. To evaluate the proposed policy model two path-checking algorithms (depth-first search (DFS) and breadth-first search (BFS)) are applied to real datasets, and the time spent on access requests is calculated in the social graph of these datasets. The results showed DFS takes less time than BFS to do the task defined.

1 Introduction

Online Social Network (OSN) provides an environment for users to interact with each other. Users who are connected to an OSN may share their sensitive and private information such as contact information, educational information, photos, and videos. Due to the expansion of OSNs, a large number of videos and photos are shared and uploaded, which necessitates further research on data security and privacy [1–4].

Access control methods are used to keep information from unauthorized access. Traditional access control methods such as Discretionary Access Control (DAC) [5], Mandatory Access Control (MAC) [6], and Role-based Access Control (RBAC) [7–9] are not suitable for social networks because access control in these methods is based on the object and subject specification. The authors in [10] believe that traditional methods do not protect information when users make a mistake. As social networks generally contain a large number of users, it is very difficult to recognize the authorized ones. Therefore, traditional methods fail to manage information access in these networks.

Access control in OSN is based on the relationships between users, while this has not been considered in traditional access control. In relation-based access control, a user can communicate with another user without being aware of the user's namespace. Some-

* Corresponding author.
Email addresses: fatemeh.nazerian@qaemiau.ac.ir (F. Nazerian), motameni@iausari.ac.ir (H. Motameni)
https://dx.doi.org/10.22108/JCS.2023.136750.1120
ISSN: 2322-4460

times, in a social network, data is shared based on events such as location and text instead of considering the relationship between the owner and the requestor. Relationship-based access control cannot support textual or global information. For more details, consider this example: Bob publishes a photo and wants those who work in company 'A' not to be able to see his photo, or Dave goes on a journey and takes a picture of his current position and he wants only the people in this city see this picture. As ReBAC relies on the type and depth of the relationship but there is no information about users and relations, it cannot formulate the above scenario. In [11–13], Attribute-based access control (ABAC) was integrated with ReBAC to create a flexible access control method; however, this method lacked a strong policy language for expressing access control policies (these researchers used a regular expression to formalize access control policy).

In this paper, the relation-based access control is extended. User attributes and relation attributes are added to introduce a new attribute relation-based access control model (A-ReBAC). To achieve efficient access control, a policy model is proposed for the new access control model. Policies are categorized into two classes: 1) User policies and 2) System policies, the user policy was illustrated based on attributes of users, the relationship between users, and the trust value of the relation. Conflict and priority were illustrated in the system policy. Then, an appropriate and unambiguous policy language is needed to formulate the formal specification of the access control policy model. Hybrid Logic is suited to this goal because it can specify access control policy from the viewpoint of the user and represent a much richer form of relationship. This paper focuses on Hybrid Logic to formulate a policy model of attribute relation-based access control that is uncovered yet (In [14], hybrid logic is used for expressing relationship-based access-control policies). Then to show the performance of the proposed approach, two real datasets are considered and some attributes are added to two datasets to support our proposed approach. Next, two path-checking algorithms (depth-first search and breadth-first search) are used to verify the time spent on the requests in the social graph of the proposed approach.

The structure of this paper is organized as follows. Section 2 provides related work. In Section 3, the proposed model is illustrated. In Section 4, Hybrid Logic is introduced as a policy language, and in Section 5, user policies are categorized and Hybrid Logic is extended to support user policy. In Section 6, system policy is illustrated, and Hybrid Logic is extended to support it. The proposed approach is implemented with a real dataset in Section 7. In Section 8, the proposed approach is compared with some access control models already introduced in the literature. Finally, Section 9 concludes the paper.

2 Related Works

In ReBAC, access to information depends on the relationships between users [15, 16]; thus, ReBAC can be used for access control policy in OSN. The researchers in [17, 18] improved the access control policy in OSN and added trust level to the relationship between the owner and requester, which indicates the level of relationship between them, and authorized users were distinguished in terms of the type, depth, and trust level of the relationship between users. The authors in [19] proposed a trust-based access control, called STBAC, for social networks. In STBAC, access to data is based on trusted computing that provides access to data for users. While Albladi et al. [20] assumed trust may cause higher susceptibility to cyber-attacks.

Access control in existing work is either too restrictive or too loose; thus, an access control model based on semantics for online social networks was proposed to increase viable protection [21]. In [21], the type of resource (e.g., photo, video, etc.) is considered to allow/reject an access request. Thus, the access control rules should be defined for each resource type, and the access control policies should be applied as a whole to the object or resource. In [22], a dynamic, transparent, and privacy-driven access control mechanism was proposed for textual messages published in OSN to cover the lack of a method in [21]. As the trust level cannot be measured, non-trusted users may receive sensitive information. The authors in [23] proposed a novel spatiotemporal access control for online social networks and focused on considering temporal and spatial factors comprehensively.

Cheng et al. in [24], proposed an access control model to support the user-to-user (U2U), user-to-resource (U2R), and resource-to-resource (R2R) relationships, and authorization policies are defined in terms of patterns of relationship paths on the social graph. In [11], the UURAC model in [24] was extended to enable attribute-aware access control. Then, in [25], a new U2U relationship-based access control model, called UURAC, was proposed, and path-checking algorithms were used to determine whether the required relationship path between users for a given access request exists. Regular expression was used as a policy language in [11, 24, 25].

With the rapid development of OSN, ReBAC cannot meet the users' requirements because sometimes textual information is needed for access control policies. ABAC was applied in [26] to fine-grained access controls; thus, the properties of users, sources,

and environment were considered [12, 27, 28]. In [11–13], Attribute-based access control (ABAC) was integrated with ReBAC to create a flexible access control method; however, this method lacked a strong policy language for expressing access control policies. In [29], real-world entities such as the state, historical events, and public figures were modeled as public information, which allows users to find other users in their respective fields of work. Two graphs (i.e., user graph and public information graph) and the relation between the two graphs were considered in this model. In the research, there are many links between users and public information; therefore, finding appropriate links between users and public information is important and this method will be more complicated to find all links.

Most researchers used regular expressions to formalize access control policy [11, 13, 25, 30, 31]. Fong et al. [15] introduced a formal model using modal language to define access control policies in OSN. Researchers in [32] introduced Hybrid Logic and combined modal language with features and additional operations. In [33], hybrid logic was used to express the access control policy in collaborative systems. Authors in [14] improved the model formalized in [15] to provide better performance in assessing access control policies in OSN. In [14, 15], hybrid logic was used for expressing relationship-based access-control policies. In our paper, ABAC and ReBAC are integrated into a new model, called A-ReBAC, to achieve efficient access control, and a new policy model is introduced based on A-ReBAC, then Hybrid Logic is used for expressing attribute relationship-based access-control policies.

3 Attribute Relational Based Access Control (A-ReBAC) Model

This section develops the foundations of the A-ReBAC model, which include the social graph and access control components.

3.1 Social Graph

The A-ReBAC model is a combination of ReBAC and ABAC. This model includes users, attributes of users, relationships between users, and attributes of relations. A sample of the social graph is shown in Figure 1.

The social graph is modeled as $G = \langle U, E \rangle$.

$U = \{\langle u, \text{user-attr} \rangle \mid u \in U, \text{user-attr} \in P_U\}$

$E = \{\langle e, \text{relu-attr} \rangle \mid e \in N * N, \text{relu-attr} \in P_E\}$

where U is a set of users, which is divided into two categories: Access User (AU) referring to users who request access, and Target User (TU) referring to those who receive an action; and PU is a set of user attributes such as name, age, hobbies, location, and carrier (user attributes can be classified as basic features, environment features, and organizational affiliations); E is a set of edges in the social graph, which represents the relationships between users in OSN; and PE is the relation attributes such as relation type and trust.

3.2 Components of the Policy Model

Figure 2 shows the policy of the proposed approach and includes user and system policies.

Policies are categorized into two classes:

(1) User policies that include a policy of access user (P-AU) and policy of target user (P-TU) defined by the user.
(2) System policies (PS) that refer to the policies defined by the system.

The social graph contains users and the relationships between them. Users and their relationships have their attributes.

4 Hybrid Logic as Policy Language

In [18], hybrid logic is used to determine the policy of access control based on the relationship. In the following, the syntax and semantics of Hybrid Logic are illustrated. The syntax is based on four sets: Nom represents a set of nominal symbols, which contain the same nodes in the social network graph and can be a user; Var is a set of variables; R is a set of relationships that include the relation between users or nodes; and P is a set of propositional symbols, which is considered for attributes of users or nodes in the social graph. If $n \in Nom$, $p \in P$, $r \in R$, $x \in Var$, then the HL syntax of the proposed approach is as follows:

$$t ::= n \mid x$$
$$\Phi ::= t \mid p \mid \neg\Phi \mid \Phi_1 \wedge \Phi_2 \mid \langle \alpha_i \rangle \Phi \mid @_t \Phi$$

$@_t$ is the world from the viewpoint of t.

The M model is considered for the proposed approach and is a triple relationship as $(U, \{R \subseteq U \times U\}, V)$, including a non-empty set of users (U), a binary relationship of R (which shows the relationships between users), and $V : (Nom \cup P) \rightarrow 2^U$. The semantics of the HL formula is defined by the satisfaction relation $M, u, g \models \Phi$, where $u \in U$ and $g : \text{Var} \rightarrow U$ is a function from variables to users. In the satisfaction relation $M, u, g \models \Phi$, the formula Φ under g is in

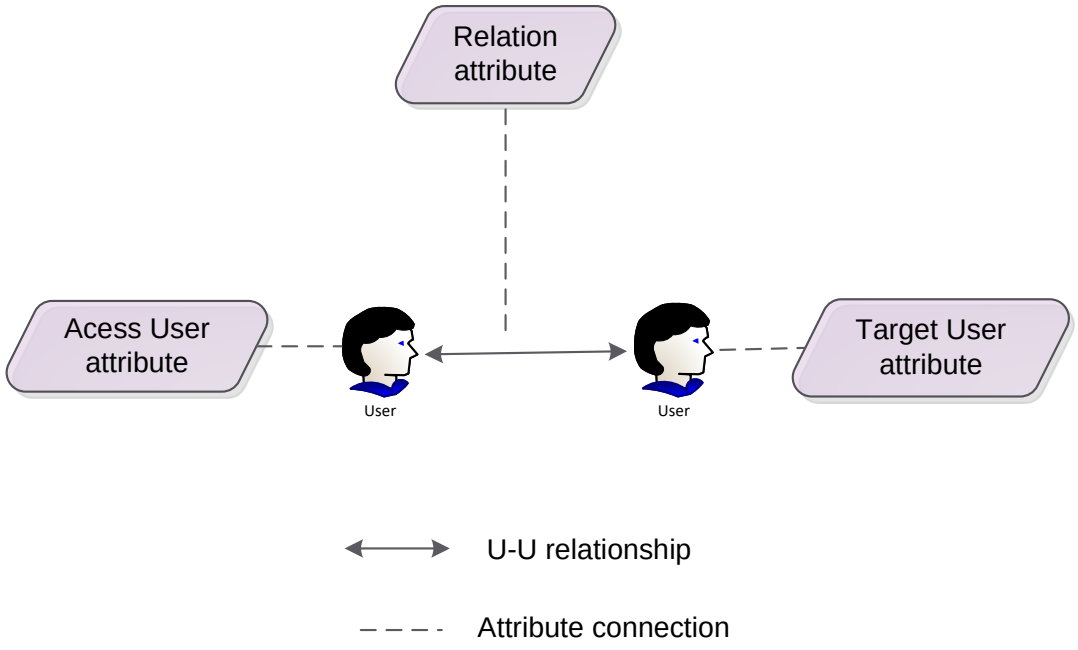


Figure 1. A Sample of a Social Graph.

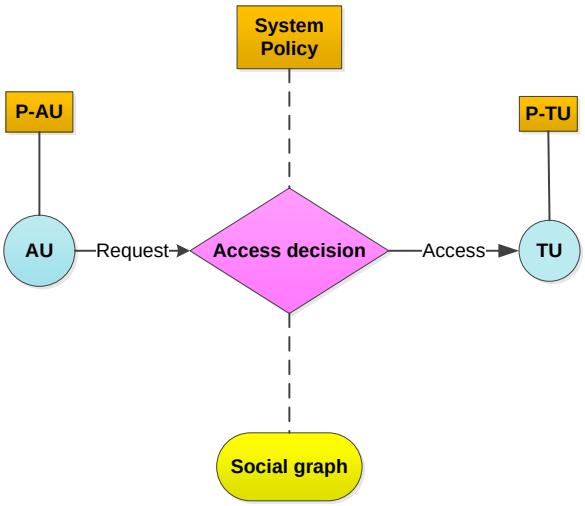


Figure 2. Access Control Policy.

the node u of the model M. The satisfaction relation in HL is defined as follows:

$$
\begin{aligned}
&M,u,g \models x && \equiv u = g(x)\\
&M,u,g \models n && \equiv V(n) = \{u\}\\
&M,u,g \models p && \equiv u \in V(p)\\
&M,u,g \models \neg\Phi && \equiv M,u,g \not\models \Phi\\
&M,u,g \models \Phi_1 \wedge \Phi_2 && \equiv (M,u,g \models \Phi_1) \text{ and } (M,u,g \models \Phi_2)\\
&M,u,g \models \langle\alpha_i\rangle\Phi && \equiv (M,u,g \models \Phi) \text{for some} (u,u')\\
&\text{in relation } \alpha_i &&\\
&M,u,g \models @_n\Phi && \equiv M,u_a,g \models \Phi \text{ where } V(n) = \{u_a\}\\
&M,u,g \models @\Phi && \equiv M,g(),g \models \Phi
\end{aligned}
$$

In access control scenarios, two elements are considered: users (owner and requester) and access control policy.

- The owner and requester are users in the social graph, which are considered free variables of 'own' and 'req', respectively.
- In the proposed model, two kinds of policies are considered: system policy and user policy. User policies specify how to release the user's resources, and system policies determine the access control on users and resources. Access control policies are expressed as a formula (Φ) in Hybrid Logic. HL formula is defined by $M,u,g \models \Phi$ satisfaction relation, where the formula Φ under g is in the node u of the model M. In the following, user and system policies are described in detail.

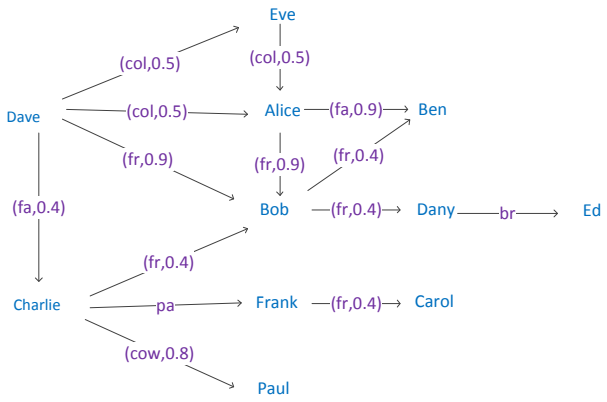


Figure 3. Social Graph.

5 User Policy

Users can determine policies for their resources (photos, videos, etc.) based on the attributes of users, the relation between users, and trust value. To show the usage of the proposed approach, the sample social graph in Figure 3 is considered. Relation attributes are shown in the social graph (Figure 3) as relation type and trust value. The relation is explained in Table 1, and the trust value is a number between 0 and 1, which shows trust between users.

For example, $(\text{col}, 0.5)$ between Dave and Eve in Figure 3 means the relation type between Dave and Eve is Colleague (according to Table 1), and the value of trust is 0.5. The first column in Table 1 describes the type of relation, and the second column shows some profile attributes of users in Figure 3.

To show the expressiveness of the proposed approach, several real-life scenarios are considered, and access control policies are formulated using HL (own, req), including own and req as free variables and formulas with satisfaction relations. Operator $@own$ is used for policy from the owner's viewpoint.

Table 1. Relation and Profile Attributes of Figure 3.

Relation Attribute		**Profile Attribute**
Symbol	Description	Name
Col	Colleague	Birthday
Fr	Friend	Company
Fa	Family	Colleague
Cow	Co-worker	School Name
Pa	Parent	Hobby
Br	Brother	Sport

5.1 User Policy Based on the Users' Attributes

When a user sets a policy based on attributes, the proposition in Hybrid Logic is used to determine the user's policy. The syntax of Hybrid Logic is as follows:

$$t ::= n|x$$
$$\Phi ::= t|p|\neg\Phi|\Phi_1 \wedge \Phi_2| < \alpha_i > \Phi|@_t\Phi$$

The semantic is d as follows:

$$M, u, g \models p \equiv u \in V(p)$$

Examples of these policies are shown below:

Scenario 1: In Figure 3, Alice uploads a file, but she prefers that only the students at college 'A' can have access to the file. This policy is as follows:

$$@_{\text{Alice}}(\text{incollegeA}(\text{req}))$$

Where $@_{\text{Alice}}$ means the policy from Alice's viewpoint. Users in social graphs save their information to protect their resources, and the proposition symbol P is used to verify the user attributes. With the above policy, users who save college 'A' in their profile can access Alice's file.

Scenario 2: Dave prefers that only those who are in the city 'A' can see his image:

$$@_{\text{Dave}}(\text{incityA}(\text{req}))$$

With this policy, users whose saved cities match 'A' can see Dave's image.

Scenario 3: Charlie sends a file and prefers that only people who are over 15 years old can see it.

$$@_{\text{Charlie}}(\text{ageover15}(\text{req}))$$

With this policy, users whose birthdates are before 2004 can see the file.

5.2 User Policy Based on the Relations Between Users

In this case, the relation in Hybrid Logic is used to formulate user policy. The syntax of Hybrid Logic is as follows:

$$t ::= n|x$$
$$\Phi ::= t|p|\neg\Phi|\Phi_1 \wedge \Phi_2| < \alpha_i > \Phi|@_t\Phi$$

The semantic is d as follows:

$$M, u, g \models \langle\alpha_i\rangle\Phi \equiv M, u, g \models \Phi \text{ for some } (u, u') \text{ in relation } \alpha_i$$

To illustrate this policy, consider the following scenarios:

Scenario 4: In Figure 3, Dave uploads a picture

and he prefers that only his family members can see the picture. In this policy, the relation $< \text{family} >$ in Hybrid Logic is used as follows:

$$@_{\text{Dave}}(< \text{family} > \text{req})$$

Using Hybrid Logic, we can support k-common friends [34]. For example, $< \text{friend} > 3$ shows at least three different friends. An example is shown in scenario 5.

Scenario 5: In Figure 3, the assumption is that Charlie uploads a photo that is authorized for both Charlie's friend and Charlie's friend of a friend. This policy is formulated below:

$$@_{\text{Charlie}}(< \text{friend} > \text{req} \vee < \text{friend} >< \text{friend} > \text{req})$$

To set a policy, proposition, and relation can be combined. This composition is shown in scenario 6.

Scenario 6: In Figure 3, Bob prefers that only his friends who do not work at the rival company can have access to his photo.

$$@_{\text{Bob}}(< \text{friend} > \text{req} \wedge \neg \text{incompanyB(req)})$$

Bob's policy is determined by the conjunction of relation and proposition. In Figure 3, users who have a *friend* relation with Bob and do not work at company 'B' can have access to Bob's photo.

5.3 User Policies Based on Trust Value

The relations between users are assigned a trust value that shows trust between users. To consider the trust value in our approach, the syntax of Hybrid Logic is extended with $T < \alpha_i >$ added, which means the trust value of the relationship between users. The syntax of Hybrid Logic is defined as follows:

$$t ::= n|x$$
$$\Phi ::= t|p|\neg\Phi|\Phi_1 \wedge \Phi_2| < \alpha_i > \Phi| < -\alpha_i > \Phi|@_t\Phi| \downarrow x\Phi|T < \alpha_i >$$

The semantics of $T < \alpha_i >$ is defined as follows:

$$M, u_a, g \models t < \alpha_i > \Phi \equiv M, u_b, g \models \Phi \text{ for } (u_a, u_b) \in \alpha_i \text{ where } T < \alpha_i > \geq t$$

With this policy, users can share their resources with trusted users. This policy is shown in scenario 7.

Scenario 7: In Figure 3, Bob sends a file. He wants that only the users who have a relationship *friend* with him and the trust value of the relationship is higher than 0.8 can have access to the file. This policy is formulated using Hybrid Logic as follows:

$$@_{\text{Bob}}(0.8 < \text{friend} > \text{req})$$

With this policy, Alice and Dave can access the file.

6 System Policy

The administrator defines the system-level policies to protect users and the relationship between them. In this work, system policies are verified in two aspects: conflict policy and priority relation.

6.1 Conflict Policy

When a resource has multiple conflicting policies, the system decides about it. Conflict occurs when two users set conflicting policies for a resource, where the satisfaction of all policies is not possible. The syntax of Hybrid Logic is as follows:

$$t ::= n|x$$
$$\Phi ::= t|p|\neg\Phi|\Phi_1 \wedge \Phi_2| < \alpha_i > \Phi|@_t\Phi$$

The semantic is d as follows:

$$M, u, g \models \Phi_1 \wedge \Phi_2 \equiv (M, u, g \models \Phi_1) \text{ and } (M, u, g \models \Phi_2)$$

An example is given in the following scenarios.

Scenario 8: Consider a sample social graph in Figure 3. Dave shares a photo and sets a policy according to which only his friends or colleagues can see the photo. Alice is one of them. Alice sets a policy, based on which only Bob sees the photo. According to Dave's policy, Eve, Alice, and Bob can see the photo, but according to Alice's policy, only Bob can see the photo.

The system uses two methods to solve the conflict policies problem: combining policies with conjunction and the preceding method.

a) In combining policy, the system uses conjunction. For example, Scenario 8 is as follows:

$$@_{Dave}(< friend > req \vee < college > req) \wedge @_{Alice}(NameisBob(req))$$

With the conjunction of two policies, only Bob can see the photo. In this state, Dave's policy is ignored.

b) In the preceding method, the first user who sets a policy has priority over the others. In Scenario 8, Dave's policy has priority over Alice's.

6.2 Priority Method

Using the priority method, the system considers the hierarchy for the relationship in the social graph and decides across the relation hierarchy to access control. An example of the relationship hierarchy is shown in Figure 4. In this figure, the relation 'parent' has a higher priority than 'family' and 'close friend'. The relationship hierarchy suggested in [29] is used in the current study for the system policy.

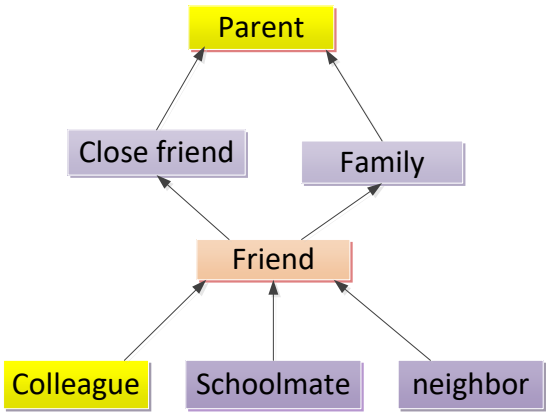


Figure 4. Hierarchy Between Relations.

Relationship hierarchy is defined by user relation, and the operator $\leq$ is defined by the relationship. Consider α_i and α_j as two relations in the social graph, and $\alpha_i \leq \alpha_j$; then relation α_j has priority over relation α_i. The operator $\leq$ is reflexive, anti-symmetric, and transitive. To consider a priority, the syntax of HL is extended as follows:

$$t ::= n|x$$
$$\Phi ::= t|p|\neg\Phi|\Phi_1 \wedge \Phi_2| < \alpha_i > \Phi| < -\alpha_i > \Phi|@_t\Phi$$
$$|\downarrow x\Phi|T < \alpha > |\lceil\langle\alpha_i\rangle\rceil Phi$$

The semantics $\lceil\langle < \alpha_i > \rangle\rceil\Phi$ is defined as follows:

$$M, u_a, g| = \lceil\langle\alpha_i\rangle\rceil\Phi \equiv M, u_b, g| = \Phi \text{ for } (u_a, u_b) \in \alpha_j \text{ where } \alpha_i \leq \alpha_j$$

To make it clearer, consider Scenario 9 with the relation hierarchy in Figure 4.

Scenario 9: Dave shares a picture and decides that only his friends can see it.

$$@_{\text{Dave}}(\lceil\langle\text{friend}\rangle\rceil(\text{req}))$$

Under this policy, the users who have relationships with parents, close friends, and family can see this picture. With this method, the parent's policies have priority over the child's.

7 Evaluation

To demonstrate the scalability of the proposed approach, two different real datasets [35] were utilized to evaluate the A-ReBAC model. The path search algorithms were applied to these datasets, and all experiments were implemented in MATLAB on a machine with Dual Core at 3.0 GHz and 4 GB of memory running the Windows 7 operating system.

7.1 Dataset

Two datasets suggested in [35], namely Soc-epinions 1 and soc-Slashdot0902, were considered in this study. They consist of 75,879 (508,837) and 82,168 (948,464) nodes (edges), respectively.

To support the node attribute, the original dataset was modified, and some information had to be added manually. Attributes such as username, gender, career, date of birth, and hometown were considered profile attributes and assigned to each node (user). A unique name was chosen for each node, ensuring differentiation from other nodes. Gender for each node was randomly selected from 0 or 1, where 0 denotes male and 1 denotes female. Each node was randomly assigned 20 different careers. Date of birth was randomly selected between 1927 and 2007 for each node. Finally, each node was randomly assigned to 20 different cities. To support the relation attribute, a random value between 0 and 1 was considered for each relation, indicating the trust value between users. This paper focuses on one type of relationship.

7.2 Time Analysis for Relationship Attribute

As mentioned in Section 7.1, a random value between 0 and 1 is considered for the relation attribute, representing the trust value between users. Two algorithms, depth-first search and breadth-first search, were applied in the experiments. The time required for the source node (user) to reach the target node (user) was computed for ten pairs of source and target nodes. Subsequently, the average time was calculated for depths ranging from 1 to 4. The results of the first experiment for datasets soc-Slashdot0902 and Soc-epinions1 are shown in Tables 2 and 3, respectively. The trust value in this experiment is set to be higher than 0.5. The comparison of the two algorithms for each dataset is illustrated in Figures 5a and 5b. As demonstrated in Figure 5a and 5b, the depth-first algorithm exhibits shorter execution times compared to the breadth-first algorithm.

7.3 Time analysis for User Attribute

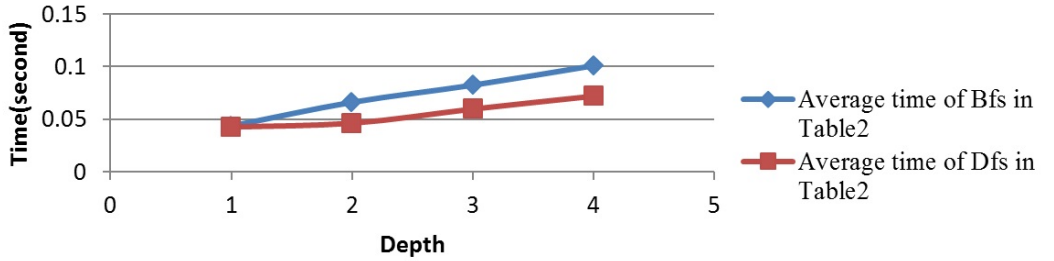
In the second experiment, Scenario 3 in Section 5.1 was taken into consideration. In the scenario, people who were over 15 years old were considered. The results of the experiment are shown for dataset soc-Slashdot0902 and dataset Soc-epinions1 in Tables 4 and 5, respectively. In this experiment, the required time of the source node to the target node was computed for ten pairs of source and target node; then, the average time is computed with depths from 1 to 4. The results of two datasets soc-Slashdot0902 and Soc-epinions 1 are compared in Figures 6a and 6b, re-

Table 2. Time analysis of relation attribute with Dfs and Bfs algorithm for Soc-Slashdot0902; ten pairs of source and target node are considered; then, the average time is computed with depths from 1 to 4.

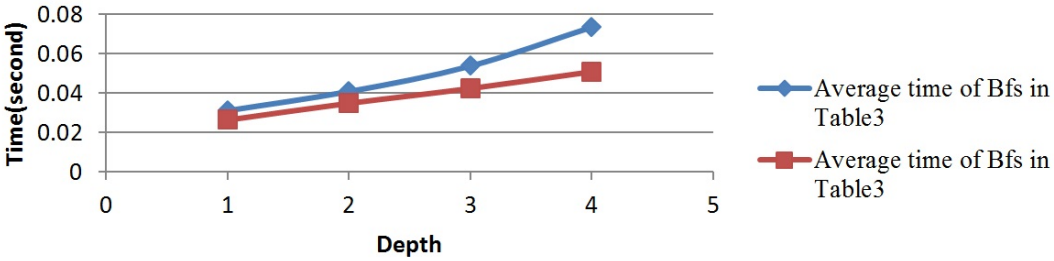
	DFS				BFS			
Node	Depth=1	Depth=2	Depth=3	Depth=4	Depth=1	Depth=2	Depth=3	Depth=4
Average	0.0427435	0.0464108	0.0598342	0.0720981	0.0437562	0.0662818	0.0825338	0.1009991

Table 3. Time analysis of relation attribute with Dfs and Bfs algorithm for Soc-epinions1; ten pairs of source and target node are considered; then, the average time is computed with depths from 1 to 4.

	DFS				BFS			
Node	Depth=1	Depth=2	Depth=3	Depth=4	Depth=1	Depth=2	Depth=3	Depth=4
Average	0.0263472	0.0348283	0.0423231	0.0507858	0.0311097	0.0407791	0.0536638	0.0734368



(a) Comparison DFs & BFs Algorithm for Soc-slashdot0902



(b) Comparison DFs & BFs Algorithm for Soc-epinion1

Figure 5. Average time with trust greater than 0.5 for dataset soc-slashdot0902 and soc-epinion1 in Tables 2 and 3, respectively.

spectively. As shown in Figure 6, the depth-first algorithm has less time than the breadth-first algorithm.

7.4 Verification of Time Analysis

As mentioned in Section 3, the social graph is modeled as $G = \langle U, E \rangle$, where U is a set of users and E is a set of edges in the social graph, representing the relationships between users in OSN. PU is a set of user attributes, such as name, age, hobbies, location, and carrier, while PE is the relation attributes, such as relation type and trust.

In the analysis of relationship attributes for user u_a, two following conditions are examined:

a) All binary relations $(u_a, u_b) \in \alpha_i$ where $T < \alpha_i >\geq$ 'Trust Value' or

b) All binary relations $(u_a, u_b) \in \alpha_i$ where $PE =$ 'Relation Type'.

In the analysis of user attributes for user u_a, the following condition is examined:

c) All users u_b for which $(u_a, u_b) \in \alpha_i$ then $PU_b =$ 'User Attribute'

Trust Value, Relation Type, and User Attribute are constant values used to search for the desired user in the social graph.

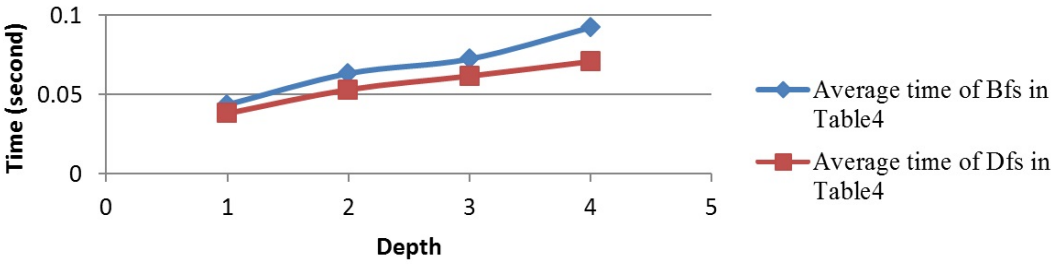
In DFS, a path from the starting node to the ending node is followed, then another path, in the same way, is followed, and it continues until all nodes are visited. In BFS, nodes are visited level by level. All nodes on one level are visited before moving to the next level; thus, if our tree is wide, DFS is better, while if our

Table 4. Time analysis of node attribute with Dfs and Bfs algorithm for soc-slashdot0902; ten pairs of source and target node then are considered, and the average time is computed with depths from 1 to 4.

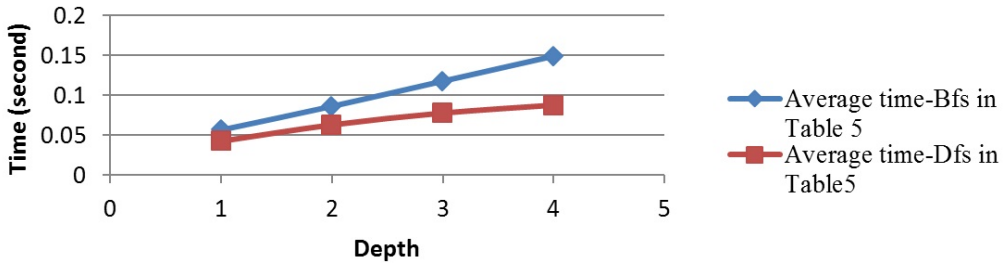
	DFS				BFS			
Node	Depth=1	Depth=2	Depth=3	Depth=4	Depth=1	Depth=2	Depth=3	Depth=4
Average	0.0380684	0.0530196	0.0615925	0.0707093	0.0435745	0.0632009	0.0723068	0.0922077

Table 5. Time analysis of node attribute with Dfs and Bfs algorithm for soc-epinion1; ten pairs of source and target node then are considered, and the average time is computed with depths from 1 to 4.

	DFS				BFS			
Node	Depth=1	Depth=2	Depth=3	Depth=4	Depth=1	Depth=2	Depth=3	Depth=4
Average	0.042495	0.0628879	0.0776796	0.0875503	0.0565425	0.0861186	0.1175564	0.1488434



(a) Comparison DFs & BFs Algorithm for Soc-epinion1



(b) Comparison DFs & BFs Algorithm for Soc-epinion1

Figure 6. Average time with user attribute for dataset soc-slashdot0902 and soc-epinion1 in Table 4 and 5 respectively.

tree is very deep, BFS is better.

For example, dataset Soc-epinions1 has 75.879(508.837) nodes (edges). The number of nodes is denoted by $N(G)$, and the number of edges with $N(E)$. For the Soc-epinions1 dataset, $N(G) = 75.879$ and $N(E) = 508.837$. In this dataset, the node is distributed in level and the graph is very wide; therefore, BFS needs much memory, and the time spent is longer than that of DFS. On the other hand, in our experiment, the maximum depth is 4; thus, the tree is not very deep, and DFS is better for this experiment.

8 Comparison

In this section, our proposed approach is compared with the access control model proposed previously in the literature and is shown in Table 6.

The approach proposed in [13] considered the attribute in ReBAC and proposed a hybrid access control model (HAC). They designed a new policy specification language to specify policies based on attributes and relationships. In [13], policies are evaluated according to the paths between the access requester and target user in the social graph; thus, there is not a verified formal and accurate evaluation and system policy.

Authors in [29] presented the access control policy based on Hybrid Logic, but attributes were considered as a public graph, and a new connection between the user graph and public information was considered. In their study, category relation among public information and the relationship hierarchy were in-

Table 6. Comparison of Access Control Models for OSN.

	Shan et al. [13]	Pang et al. [29]	Cheng et al. [25]	Cheng et al. [11]	This Paper
User Attributes	✓	✓		✓	✓
Relation Attributes	✓	✓		✓	✓
U2U Relation	✓	✓	✓	✓	✓
U2R Relation				✓	
System Policy					✓
Conflict Resolution		✓	✓		✓
Trust		✓			✓
Policy Language	Regular Expression	Hybrid Logic	Regular Expression	Regular Expression	Hybrid Logic

troduced, and conflict was called and verified as collaborative access control, and to increase the reliability of user relationship, trust value was added to the user graph. However, the approach proposed in [29] contains a user graph and public information graph; therefore, there are many links in this approach, and finding appropriate links between users and public information is complicated, and the approach lacks scalability.

The approach suggested in [25] introduced the user-to-user relationship-based access control (UURAC), and policy specification was determined with regular expression. Access control policies in their work include requested action, multiple relationship types, the starting point of evolution, and the number of hops on the path. To resolve conflict, three simple approaches, i.e., conjunctive, disjunctive, and prioritized were considered. In their research, attributes of users were not taken into account.

In [11], attributes were integrated with the relationship access control, and attribute-based policies were expressed based on path patterns between the accessing user and the target user. In this work, the user-to-resource relation was considered, but the conflict and system policies were ignored. In [11], policy language is based on regular expression while Hybrid Logic is much richer in this goal because it can specify access control policy from the viewpoint of the user. In this paper when a user requests access to a resource, the proposed model relies on the relationship between the access user and resource' owner as the target user, thus in this paper, U2R access is based on the U2U relationship.

9 Conclusions

In this paper, first, the relationship access control was extended. User attributes and relation attributes were added to introduce the A-ReBAC model. With adding attributes to the social graph, the flexibility of ReBAC increased because users can determine policies for their resources (photo, video, etc.). These policies are based on the attributes of users, the relation between users, and trust value. These policies increased the security of users' resources.

Second, the user policy was illustrated based on the attributes of users, the relationship between users, and the trust value of the relation. Conflict and priority were illustrated in the system policy; then, user and system policy was expressed based on Hybrid Logic, and Hybrid Logic was extended to support system and user policy. The Hybrid Logic was used because it is an accurate language and can specify access control policy from the user's viewpoint and represent a much richer form of relationship.

Third, Real datasets were used to evaluate the performance of the proposed approach. Two graph search algorithms (i.e., BFS and DFS) were used to find appropriate nodes or users in the social graph. The path-checking algorithm was used because the relationship-based access control is shown with a social graph. The path-checking algorithms are important for searching nodes in the graph. The results showed that DFS takes less time than BFS.

To secure access control, attributes of the users should be filled in exactly. In the future, we plan to design a graphical user interface (GUI) to inform users how to enter the minimum data needed for enhancing security. We will extend the social graph in the proposed A-ReBAC model to support the user-to-resource and resource-to-resource relationships.

References

[1] K. Shah and D. Patel. Exploring the access control policies of web-based social network. In *ICDSMLA 2019: Proceedings of the 1st International Conference on Data Science, Machine Learning and Applications*, pages 1614–1622. Springer, 2020. doi:10.1007/978-981-15-1420-3_168.

[2] Hongyu Gao, Jun Hu, Tuo Huang, Jingnan Wang, and Yan Chen. Security issues in online social networks. *IEEE Internet Computing*, 15 (4):56–63, 2011. doi:10.1109/MIC.2011.50.

[3] J. H. Park, Y. Sung, P. K. Sharma, Y. S. Jeong, and G. Yi. Novel assessment method for accessing private data in social network security services. *The journal of supercomputing*, 73:3307–3325, 2017. doi:https://doi.org/10.1007/s11227-017-2018-6.

[4] R. Ghazal, A. K. Malik, N. Qadeer, B. Raza, A. R. Shahid, and H. Alquhayz. Intelligent role-based access control model and framework using semantic business roles in multi-domain environments. *IEEE Access*, pages 12253–12267, 2020. doi:10.1109/ACCESS.2020.2965333.

[5] S. Osborn, R. Sandhu, and Q. Munawer. Configuring role-based access control to enforce mandatory and discretionary access control policies. *ACM Transactions on Information and System Security (TISSEC)*, 3(2):85–106, 2000. doi:10.1145/354876.354878.

[6] S. Osborn. Mandatory access control and role-based access control revisited. In *RBAC '97 Proc. of the 2nd ACM Workshop on Role-Based Access Control*, pages 31–40, 1997.

[7] R. Sandhu, E. Coyne, H. Feinstein, and C. Youman. Role-based access control models. *IEEE Computer*, page 38–47, 1996. doi:10.1109/2.485845.

[8] F. Nazerian, H. Motameni, and H. Nematzadeh. Secure access control in multidomain environments and formal analysis of model specifications. *Turkish Journal of Electrical Engineering and Computer Science*, 26(5):2525–2540, 2018. doi:10.3906/elk-1802-55.

[9] F. Nazerian, H. Motameni, and H. Nematzadeh. Emergency role-based access control (e-rbac) and analysis of model specifications with alloy. *Journal of Information Security and Applications*, 45:131–142, 2019. doi:https://doi.org/10.1016/j.jisa.2019.01.008.

[10] D. T. Tran, D. K. Tran, and J. Kung. Interaction and visualization design for privacy interface on online social network. *SN Computer Science*, 1 (5), 2020. doi:10.1007/s42979-020-00314-9.

[11] Y. Cheng, J. Park, and R. S. Sandhu. Attribute-aware relationship-based access control for online social networks. In *Data and Applications Security and Privacy XXVIII: 28th Annual IFIP WG 11.3 Working Conference, DBSec 2014, Vienna, Austria, July 14-16, 2014. Proceedings 28*, pages 292–306. Springer, 2014. doi:https://doi.org/10.1007/978-3-662-43936-4_19.

[12] Z. Zhang, L. Han, C. Li, and J. Wang. A novel attribute-based access control model for multimedia social networks. *Neural Network World*, (6):543–557, 2016. doi:10.14311/NNW.2016.26.031.

[13] F. Shan, H. Li, F. Li, Y. Guo, and B. Niu. Hac: Hybrid access control for online social networks. *Security and Communication Networks*, 2018. doi:10.1155/2018/7384194.

[14] G. Bruns, P. W. Fong, I. Siahaan, and M. Huth. Relationship-based access control: Its expression and enforcement through hybrid logic. In *Proc. Second CODASPY*, pages 117–124, 2012.

[15] P. W. L. Fong and I. Siahaan. Relationship-based access control policies and their policy languages. In *Proceedings of the ACM Symposium on Access Control Models and Technologies (SACMAT)*, pages 51–60, 2011.

[16] C. E. Gates. Access control requirements for web 2.0 security and privacy. In *Proceedings of IEEE Web 2.0 Privacy and Security Workshop (W2SP'07)*, Oakland, California, May 2007.

[17] B. Carminati, E. Ferrari, and A. Perego. Rule-based access control for social networks. In *On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops: OTM Confederated International Workshops and Posters, AWeSOMe, CAMS, COMINF, IS, KSinBIT, MIOS-CIAO, MONET, OnToContent, ORM, PerSys, OTM Academy Doctoral Consortium, RDDS, SWWS, and SeBGIS 2006, Montpellier, France, October 29-November 3, 2006. Proceedings, Part II*, pages 1734–1744, 2006. doi:http://dx.doi.org/10.1007/11915072_109.

[18] B. Carminati, E. Ferrari, R. Heatherly, and M. Kantarcioglu. Enforcing relationships privacy through collaborative access control in web-based social networks. In *2009 5th International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pages 1–9. IEEE, 2009. doi:10.4108/ICST.COLLABORATECOM2009.8339.

[19] S. Omanakuttan and M. Chatterjee. Trust based access control for social networks (stbac). *International Journal of Innovations in Engineering and Technology (IJIET)*, 2013.

[20] S. Albladi and G. Weir. Predicting individuals' vulnerability to social engineering in social net-

works. *Cybersecurity*, 2020. doi:10.1186/s42400-020-00047-5.

[21] B. Carminati, E. Ferrari, R. Heatherly, and M. Kantarcioglu. Semantic web-based social network access control. *Computers and Security*, 30(2-3):108–115, 2011. doi:10.1016/j.cose.2010.08.003.

[22] M. Imran-Daud, D. Sanchez, and A. Viejo. Privacy-driven access control in social networks by means of automatic semantic annotation. *Computer Communication*, pages 12–25, 2016. doi:https://doi.org/10.1016/j.comcom.2016.01.001.

[23] L. Zhang, Z. Zhang, and T. Zhao. A novel spatio-temporal access control model for online social networks and visual verification. *International Journal of Cloud Applications and Computing (IJCAC)*, 11(2):17–31, 2021. doi:10.4018/IJCAC.2021040102.

[24] Y. Cheng, J. Park, and R. S. Sandhu. Relationship-based access control for online social networks: beyond user-to-user relationships. In *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Conference on Social Computing*, pages 646–655. IEEE, 2012. doi:10.1109/SocialCom-PASSAT.2012.57.

[25] Y. Cheng, J. Park, and R. S. Sandhu. An access control model for online social networks using user-to-user relationships. *IEEE Transactions on Dependable and Secure Computing*, 13(4):424–436, 2016. doi:10.1109/TDSC.2015.2406705.

[26] E. Yuan and J. Tong. Attribute-based access control (abac) for web services. In *Proceedings of the IEEE ICWS*, pages 561–569, 2005. doi:10.1109/IRI.2012.6303043.

[27] X. Jin, R. Krishnan, and R. Sandhu. A unified attribute-based access control model covering dac, mac and rbac. In *Data and Applications Security and Privacy XXVI: 26th Annual IFIP WG 11.3 Conference, DBSec 2012, Paris, France, July 11-13, 2012. Proceedings 26*, pages 41–55. Springer, 2012. doi:https://doi.org/10.1007/978-3-642-31540-4_4.

[28] H. Shen and F. Hong. An attribute-based access control model for web service. In *2006 Seventh International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT'06)*, pages 74–79. IEEE, 2006. doi:10.1109/PDCAT.2006.28.

[29] J. Pang and Y. Zhang. A new access control scheme for facebook-style social networks. *Computer and Security*, 54:44–59, 2015. doi:https://doi.org/10.1016/j.cose.2015.04.013.

[30] S. Chakraborty, R. Sandhu, and R. Krishnan. On the feasibility of attribute-based access control policy mining. In *2019 IEEE 20th International Conference on Information Reuse and Integration for Data Science (IRI)*, pages 245–252. IEEE, 2019. doi:10.1109/IRI.2019.00047.

[31] S. Chakraborty and R. Sandhu. Formal analysis of rebac policy mining feasibility. In *CODASPY '21: Proceedings of the Eleventh ACM Conference on Data and Application Security and Privacy*, page 197–207, 2021. doi:10.1145/3422337.3447828.

[32] C. Areces and B. Ten Cate. Studies in logic and practical reasoning. *Studies in Logic and Practical Reasoning*, 2007. doi:10.1016/S1570-2464(07)80017-6.

[33] S. Damen, J. D. Hartog, and N. Zannone. Collac: Collaborative access control. In *International Conference on Collaboration Technologies and Systems (CTS)*, 2014. doi:10.1109/CTS.2014.6867557.

[34] P. Fong, M. Anwar, and Z. Zhao. A privacy preservation model for facebook-like social network systems. In *Proc. 14th European Symposium on Research in Computer Security*, pages 303–20, 2009.

[35] B. Carminati, E. Ferrari, and J. Girardi. Performance analysis of relationship-based access control in osns. In *2012 IEEE 13th International Conference on Information Reuse & Integration (IRI)*, pages 449–456. IEEE, 2012.

Fatemeh Nazerian is an IT Manager at Islamic Azad University, Qaemshahr Branch, Qaemshahr, Iran since 2019. She received her B.S. degree in Computer Engineering from Kharazmi University, Tehran, Iran, and her M.S. and Ph.D. degrees in Computer Science and Computer Engineering respectively from Islamic Azad University, Sari, Iran. Her research interests include Access Control, Security, Performance analysis, Evolutionary computation, and Formal Methods.

Homayun Motameni is an Associate Professor in the Department of Computer Engineering at Islamic Azad University, Sari Branch, Sari, Iran. He received his B.S. degree in Computer Engineering from Shahid Beheshti University, Tehran, Iran in 1995, and M.S. and Ph.D. degrees both in Computer Engineering from Science and Research Branch, Islamic Azad University, Tehran, Iran, in 1998 and 2007, respectively. His research interests include Software Engineering, Performance analysis, Evolutionary computation, Formal Method, Fuzzy systems, and Green Cloud Computing.