Geo-Social-RBAC: A Location-based Socially Aware Access Control Framework

Nathalie Baracaldo, Balaji Palanisamy, James Joshi

University of Pittsburgh
{nab62, bpalan, jjoshi} @pitt.edu

Abstract. The ubiquity of low-cost GPS-enabled mobile devices and the proliferation of online social networks have enabled the collection of rich geo-social information that includes the whereabouts of the users and their social connections. This information can be used to provide a rich set of access control policies that ensure that resources are utilized securely. Existing literature focuses on providing access control systems that control the access solely based on either the location of the users or their social connections. In this paper, we argue that a number of real-world applications demand an access control model that effectively captures both the geographic as well as the social dimensions of the users in a given location. We propose, Geo-social-RBAC, a new role based access control model that allows the inclusion of geo-social constraints as part of the access control policy. Our model, besides capturing the locations of a user requesting access and her social connections, includes geo-social cardinality constraints that dictate how many people related by a particular social relation need to be present in the required locations at the time of an access. The model also allows specification of geo-social and location trace constraints that may be used to dictate if an access needs to be granted or denied.

1 Introduction

The ubiquity of low-cost GPS-enabled mobile devices and the proliferation of online social networks allow the collection of rich geo-social information that includes the whereabouts of the users and their social connections. A number of real-world applications demand an access control (AC) model that effectively captures both the geographic as well as the social dimensions of the users in a given location. It is often possible to use this information to help restrict access to a particular set of resources given the location and social context of a user. For instance, consider a hospital where a doctor can access a patient's record if and only if the doctor is the patient's primary physician and the patient is located in the waiting room outside the doctor's office. Similarly, we may want to protect the privacy of patients by ensuring that in case a third person enters a room that is not part of the medical personnel and is not the patient's spouse, the health record should be automatically closed to avoid leaking patient's information.

In addition to geo-locations, location traces also offer interesting potential in the context of geo-social AC. In these cases, the whereabouts of a user and the people she has recently met influence how trusted the person is and the AC decision itself.

For instance, a trace-based geo-social AC policy may ensure that if a doctor was in a contagious unit, he cannot enter the new born unit unless he goes to a sanitizing facility first. It is also possible in some cases to bootstrap the trust of a user to access a resource based on the people that accompany him and the places where they have been together in the recent past. For instance, in a fast-food restaurant, a user who has just bought something should be allowed to access other areas of the restaurant such as restrooms and if she also has her kids with her, she should be allowed to use the kids' play area.

While there are many potential benefits of a geo-social AC model, unfortunately current literature does not provide a solution that allows the specification of such policies which include both geo-social as well as location traces with geo-social cardinality constraints. Most of the existing models support the specification of policies that depend on user location or other contextual factors such as time, type of device used to access the system and the type of connection used to access resources [3, 5, 16, 6, 11]. Given that many organizations use role based access control systems (RBAC) [7] to control their resources [12], several existing works have extended this model to include the location context [3, 5, 16, 11].

In this paper, we propose a fine-grained geo-social AC model, Geo-social-RBAC, that allows the inclusion of geo-social constraints as part of the AC policy. Concretely, in this paper we make the following *contributions*:

1. To the best of our knowledge, the proposed Geo-social-RBAC model is the first role based AC model that allows the inclusion of geo-social constraints as part of the AC policy.
2. Our model, besides capturing the locations of a user requesting access and her social connections, supports geo-social cardinality constraints that dictate how many people related by a particular social relation need to be present in the required locations at the time of an access. The model also allows specification of fine-grained geo-social and location trace constraints that may be used to dictate if an access needs to be granted or denied based on the historical whereabouts of users.

The remainder of this paper is organized as follows. In Section 2, we discuss the requirements of the system and present an overview of the proposed model. In Section 3, we present the components that we use as part of the system to model the location and social relations and then introduce the proposed Geo-Social RBAC. In Section 4, we present the related work and we conclude our paper in Section 5.

2 Motivation and Requirements

In this section we motivate the need for the proposed Geo-social RBAC model and present the requirements that guide the design of our geo-social AC framework. We begin by discussing the types of policies that are unique to the proposed AC model that are not supported by existing systems. Current AC models do not have the capabilities to support policies that contain geo-social traces and constraints. In this work, we focus on a RBAC [7] based geo-social model because of RBAC's well-documented advantages [12] and wide adoption. In RBAC, users and permissions are assigned to roles. In order to acquire the permissions associated with a role, a user needs to be previously assigned

to it and needs to activate it in a session. RBAC does not support location constraints and as a result, several extensions have been proposed to include location constraints [3, 5, 16, 11].

We broadly classify the existing RBAC literature into two categories namely RBAC extensions that support location based decisions [3, 5, 16, 11] such as Geo-RBAC [3] and LoT-RBAC [5] and models that extend RBAC with proximity constraints that include other user's proximity as part of the AC policies such as Prox-RBAC [10, 9]. In Table 1, we compare existing approaches based on the following types of constraints:

1. *Pure location constraints:* these constraints only take the location of the user into account, e.g., to access a confidential file, a user may need to be in a specific room.
2. *Geo-social constraints:* these constraints consider both the location and the social dimensions of the users in the policies. We further classify this type of constraints as follows. *(i) Geo-social graph-based constraints* which are based on the social graph structure, e.g., to enter into a room a person needs to be in company of at least two friends that work there and are present. *(ii) Geo-social tag-based constraints* which capture the type of relationships between the users in the social graphs in addition to the location and social constraints. For example, a child can only access a pay-to-view movie if he is in presence of his parent or a nanny.
3. *Trace-based constraints:* These constraints are based on user's trajectory and whether the user has been in contact with a particular set of individuals. We distinguish between two types of constraints. *(i) Location trace-based constraints:* which capture the past location traces of a user as part of the AC policies. For instance, consider a silicon chip manufacture company where even a minimum amount of dust may ruin an entire production batch. If an operator has been in known dusty rooms of the factory, he cannot enter the sterile chip production room unless he has previously passed through the cleaning room. This is a location trace policy as the previous whereabouts of the user determine whether he would be able to obtain the requested access. *(ii) Geo-social trace-based constraints:* which capture both the location history and the social dimensions of the users. For example, in a company, if a visitor has entered into the rooms used for induction of new employees accompanied by an administrator, he can also access the welcome package files and the internal directory web pages.

As shown in Table 1, existing models do not support many geo-social constraints that the proposed Geo-Social-RBAC incorporates. We further consider the following requirements for our model. The proposed AC framework should allow backward compatibility with RBAC based systems and should effectively support pure location, geo-social and trace-based constraints. The model should allow policies for different spatial granularity, e.g., it should be possible to specify if someone needs to be in a point in the space, at a door, on a room or in a floor of a building, in a city, among others.

2.1 Overview of the Proposed Geo-social RBAC Framework

In Geo-social-RBAC, the context of users is defined by the following information: the position of the user and his previous whereabouts, the proximity of the user to other users and the user's social relations with these individuals. The system consists of *users*,

Policy	RBAC extended with location [3, 5, 16, 11]	RBAC extended with proximity [10, 9]	*Our Approach:* Geo-Social-RBAC
Pure location constraints	Yes	Yes	Yes
Geo-social graph-based constraints	No	Yes	Yes
Geo-social tag-based constraints	No	No	Yes
Location-trace-based constraints	No	No	Yes
Geo-social-trace-based constraints	No	No	Yes

Table 1: Comparison of types of policies supported by RBAC based systems.

geo-social roles, *permissions* and *trace-based* and *geo-social-cardinality constraints*. In our model, users are assigned to geo-social roles and geo-social roles are assigned permissions. To acquire permissions of a geo-social role, users need to be assigned to it and activate it in a session. Geo-social roles can only be activated by a user when his contextual constraints allow it. Hence, a user can only activate a geo-social role when the current location, his previous whereabouts, his proximity to other users and their social relations satisfy the associated activation constraints.

3 Geo-Social-RBAC

In this section we present the details of the proposed Geo-Social-RBAC.

3.1 Social Relations

Modeling social relations is of key importance when specifying policies in a Geo-Social context. For this purpose and without loss of generality, we consider a single social graph that captures the various social relationships among the users. Here, we note that we could also use multiple social graphs services to obtain relevant social information. Let $\mathcal{G} = \langle V, E \rangle$ be a directed and asymmetric *Social Graph*, where V is a set of vertices and E a set of edges that represent users and their relationships, respectively. We also assume that there is a set of *tags* W used to annotate social relations. For each $e_{(i,j)} \in E$ there is a set that contains one or more tags $W_{(i,j)} \subseteq W$ that denote the type of relation between users i and j. A tag represents a specific type of social relation between two users such as a manager-employee relationship. This asymmetry between relations is necessary to ensure that some policies of interest can be specified. For example, suppose $W_{(i,j)} = \{\text{nanny, school\_mate}\}$ which shows that user i is the nanny and school mate of user j, while $W_{(j,i)} = \{\text{school\_mate}\}$. This allows us to later specify policies of the type "a child cannot access a web page if he is not in presence of his parent or a nanny".

Often, social relations have an inherent hierarchical structure. To represent such partial order, tags in W are organized in a lattice L_W. For instance, L_W may show that tags *teacher* and *parent* are greater than tag *student* while *teacher* and *parent* do not have any clear ordered relation, as it is the case when a child request to watch a movie.

We use the functions presented in Table 2 to extract relevant information from social graph $\mathcal{G}$. Policies in geo-social-RBAC include relations between a particular user and other users in the social graph. A valid social relation predicate $\mathcal{S}$ is formed by the functions previously listed and allows verification of the existence of a particular(s) social relation(s) or to verify if a social relation has certain properties.

3.2 Geo location and location traces

To model users location and their location traces in the proposed Geo-Social RBAC, we make use of the Open GeoSpatial consortium geometric model [1]. In this model,

Function	*Meaning*
$getSocialRelation : V \times V \rightarrow 2^W$	Returns the tags of a given social relation, e.g., *GetSocialRelation*($v_i \in V, v_j \in V$)= $W_{(i,j)}$.
$getSocialDistance : V \times V \rightarrow \{\mathbb{N} \cup \infty\}$	Returns the minimum number of edges between the specified vertices, e.g., for a direct social relation returns 1, for a friend-of-friend relation returns 2 and for two unconnected nodes ∞.
$superior : V \times V \rightarrow \{t, f\}$	Returns true if the first vertice, v_i, is *superior* to the second vertice, v_j given their tags $W_{(i,j)}$ and lattice L_W.
$commonNeighbors : V \times V \rightarrow \{t, f\}$	Given vertices v_i and v_j returns true if they have neihbors in common, otherwise returns false.
$kClique : 2^V \rightarrow \{t, f\}$	Returns true if the given vertices form a clique, otherwise returns false.

Table 2: Functions to extract relevant information from social graph $\mathcal{G}$.

elements in a space called *geometries* are modelled as *points*, *polygons* and *lines*. Geometries of interest are given names and are called *features*, and are defined as a tuple $\langle type, name \rangle$ where $type \in \{point, line, polygon\}$ represents the geometry type and $name$ represent the name of feature f, respectively, e.g., a polygon that represents an office may be named office-501. The set of all features of the system is denoted as $\mathcal{F}$.

Additionally, it is necessary to establish a reference space that we denote as $\mathcal{M}$ that provides the limits of the system of interest. Let $\mathcal{L}$ be a set of functions to validate the location of users that take as input the location of the user and identify if the location is as expected with respect to a particular place. $\mathcal{L}$ contains operations such as *overlap*, *touch*, *cross*, *in*, *contains*, *equal*, and *disjoint* [1] and may also contain more refined proximity functions as the ones presented in [9]. These functions serve to measure the proximity between a coordinate and a particular location and may be used to establish how far away a user is from others. While $location(u)$ provides coordinates, a function $\ell \in \mathcal{L}$ verifies logical information with respect to a feature f, e.g., function ℓ takes the current location of user u, $location(u)$, and a feature and validates if a user is standing at a particular door. Hence, a tuple $\langle f, \ell \rangle$ defines a spatial scope of interest.

Traces: The proposed Geo-Social RBAC also considers the location and geo-social traces that users generate as they move around $\mathcal{M}$. A *location trace* of a user u shows the places that he has visited. Concretely, during a period $[t_s, t_e]$ starting at t_s and ending at t_e, his *location trace* $\wp l_{(u,t_s,t_e)}$ is defined as a list $\langle \langle p_1, t_s \rangle, ..., \langle p_i, t_j \rangle, ... \langle p_n, t_e \rangle \rangle$ where tuple $\langle p_i, t_j \rangle$ shows that the user was at the location point p_i at time instance t_j.

Similarly, his *geo-social trace* $\wp g_{(u,t_s,t_e)}$ besides showing his whereabouts through time, also shows who he has frequented. We define his geo-social trace $\wp g_{(u,t_s,t_e)}$ as a list of tuples $\langle \langle p_1, U'_1, t_s \rangle, ... \langle p_n, U'_n, t_e \rangle \rangle$. Each item in the list besides containing p_i and t_j also includes $U'_i \subseteq U$ which is the set of users in proximity as per function $\ell \in \mathcal{L}$ of user u at time t_j. If at time instance t_j the system has no record of the whereabouts of user u, $p_i = \perp$.

To be able to specify trace-based policies, we define a *trace constraint* $\mathcal{Q}$ which consolidates both geo-social and location constraints in a single construction. A *trace clause* is a location constraint $c = \langle \alpha, \mathsf{T} \rangle$ or a geo-social constraint $g = \langle \beta, \mathsf{T} \rangle$ that need to be fulfilled within a period of time T. More concretely, α is defined by a tuple of the form $\langle f \in \mathcal{F}, \ell \in \mathcal{L} \rangle$ and β by a tuple $\langle f \in \mathcal{F}, \ell \in \mathcal{L}, s \in \mathcal{S} \rangle$. A location constraint is fulfilled by user u if his location trace $\wp l_{(u,t_s,t_e)}$, for $\mathsf{T} = [t_s, t_e]$, contains locations that satisfy α. Similarly, a geo-social constraint is fulfilled if $\wp g_{(u,t_s,t_e)}$ satisfies β.

Considering these definitions, $\mathcal{Q}$ is defined by the following grammar[1]: $C ::= C \wedge C \mid C \vee C \mid c \mid g$.

The previous construction allows the specification of policies where the whereabouts and the type of people that the user meets are relevant for making AC decisions. We use function *completeTrace* which takes as input a trace constraint $\mathcal{Q}$, a user u and determines if u has completed the trace by evaluating each trace clause q in $\mathcal{Q}$ and integrating the results. If the trace constraint is empty, *completeTrace* returns true.

3.3 Geo-social Cardinality Constraints

Geo-social cardinality constraints are key to specify whether the locations of a user's social relations should interfere with the access decisions. A geo-social cardinality clause is a tuple $c = \langle f, \ell, n, \mathcal{S} \rangle$ where $f \in \mathcal{F}$ is the feature where at least n social connections that comply with social predicate $\mathcal{S}$ need to be located at according to the proximity function $\ell \in \mathcal{L}$. Based on c, grammar: $C ::= C \wedge C \mid C \vee C \mid T$ and $T ::= c \mid \epsilon$, defines a *geo-social cardinality constraint* $\mathcal{C}$. We use function $peopleAt(u, \mathcal{C})$, which takes a user u and a cardinality constraint $\mathcal{C}$, to evaluate if the constraint is satisfied or not. When a cardinality constraint is empty (ϵ), $peopleAt(u, \mathcal{C})$ returns true.

3.4 Geo-Social-RBAC

With the key building blocks of our model introduced in the previous subsections, we now present the proposed geo-social aware AC model. We first introduce Core-Geo-Social-RBAC and then extend it to include role hierarchy.

Core-Geo-Social-RBAC is defined as a tuple $\langle U, R_{\mathcal{GS}}, A, O, P \rangle$. The model consists of a set of geo-social roles $R_{\mathcal{GS}}$, a set of users U, a set of actions A a set of objects O and a set of permissions defined as $P = A \times O$. Users are assigned to geo-social roles and geo-social roles are assigned permissions. We use function $authorized(u \in U)$ to obtain the set of roles that u is authorized for.

Definition 1. *A* geo-social role $r \in R_{\mathcal{GS}}$ *is defined as a tuple* $\langle SC, \mathcal{C}, \mathcal{Q} \rangle$ *where*

- SC *is a set that represents the spatial-scope of a role (places where the role can be activated). The set contains tuples of the form* $\langle f \in \mathcal{F}, \ell \in \mathcal{L} \rangle$. *When* $SC = \perp$ *the role does not have a spatial scope is specified.*
- $\mathcal{C}$ *is a geo-social cardinality constraint.*
- $\mathcal{Q}$ *is a trace constraint.*

In our model, a geo-social role without any constraint is equivalent to a standard role. Additionally, a geo-social role can be in one of two states *enabled*, or *disable*.

Definition 2. *A geo-social role* $r = \langle SC, \mathcal{C}, \mathcal{Q} \rangle \in R$ *is said to be* enabled *for user* u *if all the following conditions are fulfilled:* $r \in authorized(u) \wedge peopleAt(u, \mathcal{C}) \wedge completeTrace(\mathcal{Q}, u) \wedge \exists \langle f, \ell \rangle \in SC : \ell(location(u), f) \vee SC = \emptyset$. *Otherwise* r *is* disabled.

Henceforth, we refer to *geo-social roles* as *roles*. In the previous definition, a role r is enabled for a user u if u is assigned to r, she is in the required location and the geo-social cardinality and trace constraints are fulfilled. A user u can *activate* role r if it is enabled. When u activates r he can obtain all its privileges.

To show the expressiveness of our model, we present some examples in Table 3 that shows how our model can be used in a variety of scenarios.

[1] For simplicity grammars omit the parenthesis to avoid distracting readers from the main issues.

Pure location constraint policy: A *researcher* should be in the laboratory (fourth floor) in order to access any general files. Let r_1 be a researcher's geo-social role, with location scope $SC = \langle$*floor4, in*$\rangle$.
Geo-social cardinality constraint(for your eyes only): A *senior-researcher* can access a confidential vaccine compound formula only if he is in the confidential room by himself. Let r_2 be a senior-researcher's geo-social role, with location scope $SC = \langle$*ConfidentialRoom, in*$\rangle$ and a geo-social cardinality constraint $\mathcal{C} = \langle$*ConfidentialRoom*, in, 0, $\epsilon\rangle$.
Geo-social cardinality constraint (tag): An *assistant* in the research lab can only see files with private medical information of subjects if he is in the 4th floor and there are three researchers or senior-researchers (superiors) in the general research unit. Let r_3 be a senior-researcher's geo-social role, with location scope $SC = \langle$*floor4, in*$\rangle$, an a geo-social cardinality constraint $\mathcal{C} = \langle$*GeneralResearchRoom, in, 3, superior(u,x)*$\rangle$.
Trace constraint: A nurse needs to go to check all patients in their rooms in the last 2 hours before she can sign her electronically the round-sheet. Here, role nurse r_5 is associated with $\mathcal{Q} = (\langle\, room_1$*,in*$\rangle \wedge \ldots \wedge \langle\, room_n$*,in*$\rangle$,*2hours*$)$ and with permission sign electronically the round-sheet.

Table 3: Examples of policies that can be expressed in Geo-Social-RBAC.

Finally, we discuss Geo-Social-RBAC with Role Hierarchy. Role hierarchy [13] is a feature used by some RBAC systems in which roles are organized in a partial order. We define a Geo-Social-RBAC system as a tuple $\langle U, R_{\mathcal{GS}}, A, O, P, R_H\rangle$ that in addition to the components in the core-Geo-social RBAC, also incorporates the geo-social role hierarchy R_H. The semantics of R_H are defined as follows.

Definition 3. *Let $r_i, r_j \in R_{\mathcal{GS}}$ be two geo-social roles. r_i is said to be* senior *of r_j, written as $r_i \geq r_j$. If a user u assigned to r_i can activate r_j as long as r_j is enabled.*

In Geo-Social RBAC, a user that activates r_i does not automatically inherit the permissions of its junior roles unless those junior roles can be activated. A user that needs to acquire the permissions of a junior role would need to activate it in a session. We note that this design has several advantages. First, it ensures that all specified constraints are enforced in the system preventing and resolving policy conflicts that result when r_i and r_j are not simultaneously enabled. Also, it enforces the least privilege principle and automatically reduces the risk exposure of granting access [2].

We next discuss some related work for our Geo-social RBAC model.

4 Related Work

Several works have extended RBAC to include the context of the user such as the location and temporal constraints as part of the AC decision [3, 5, 16, 6, 11]. Unfortunately, these works do not allow the specification of geo-social constraints or location traces constraints as part of the policies. Some literature [15, 8] have proposed to include social relations constraints as part of the AC model. TMAC [15] is a model to establish policies that require team cooperation. Fong present ReRAC [8] where decisions are based on the relationship between the resource owner and the access requester. Carminati *et al.* [4] propose an AC model where policies are expressed based on user-user and user-resource relationships. In contrast, our model considers both geographical and social dimensions of the users for making access decisions. In [14], AC decisions are made based on the location of the resource owner, the resource requester and possibly other co-located individuals. Unlike our model, their model assumes that individuals own the resources and it is not based on RBAC, making it less suitable for company settings. Also, it does not consider location trace constraints as captured by our model.

Few works have explored the inclusion of geo-social context as part of AC systems [10, 9]. Prox-RBAC model [10] extends the Geo-RBAC model to include proximity of other individuals as part of the policy in indoor environments. Yet, Prox-RBAC does not allow the specification of geo-social constraints based on social graphs; in Prox-RBAC

valid proximity constraints are based on the type of role of other individuals in proximity of the access requester hold. Gupta *et. al* [9] extended Prox-RBAC by providing formal definitions to determine the proximity between locations, users, attributes and time, each of which is referred to as a realm. However, their work does not allow the specification of the type of policies presented in this paper. More specifically, *(i)* the model presented in [9] does not allow the specification of trace-based constraints that is well captured in our geo-social-RBAC model, *(ii)* unlike our model, the model in [9] does not allow the specification of latices specify partial orders between social relations and *(iii)* finally, the AC model presented in [9] does not include hybrid realm policies while our geo-social-RBAC approach does. To the best of our knowledge, the proposed Geo-Social-RBAC is the first research effort dedicated to providing a comprehensive role-based AC model that effectively captures both social and as spatial dimensions of the users considering both geo-cardinality and location-trace constraints.

5 Conclusions

In this paper, we presented a new access control model that includes geo-social factors of the users as part of the access control decision process. The proposed model allows organizations to specify their policy considering the geographic and social contexts of the access requester users as well as that of the users located near them. We have introduced the concepts of location and geo-location traces, that allow the specification of policies based on the whereabouts of users not only during the access control decision, but during a longer period of time such as their recent past. Our model is compatible with RBAC systems and we believe that it helps mitigate information exfiltration threats and helps better control how users access resources. As part of future work, we are working on devising new techniques to efficiently enforce our policy model.

References

1. Opengis simple features specification for sql, tech. report ogc 99-049. Technical report, OpenGIS Consortium, 1999.
2. N. Baracaldo and J. Joshi. An adaptive risk management and access control framework to mitigate insider threats. *Computers & Security*, 39:237–254, 2013.
3. E. Bertino, B. Catania, M. L. Damiani, and P. Perlasca. Geo-rbac: a spatially aware rbac. In *Proceedings of the tenth ACM symposium on Access control models and technologies*, pages 29–37. ACM, 2005.
4. B. Carminati, E. Ferrari, R. Heatherly, M. Kantarcioglu, and B. Thuraisingham. A semantic web based framework for social network access control. In *Proc. of the 14th SACMAT*, pages 177–186. ACM, 2009.
5. S. M. Chandran and J. B. Joshi. Lot-rbac: A location and time-based rbac model. In *Web Information Systems Engineering–WISE 2005*, pages 361–375. Springer, 2005.
6. M. J. Covington, W. Long, S. Srinivasan, A. K. Dev, M. Ahamad, and G. D. Abowd. Securing context-aware applications using environment roles. In *Proc. of the 6th SACMAT*, pages 10–20, 2001. ACM.
7. D. F. Ferraiolo, R. Sandhu, S. Gavrila, D. R. Kuhn, and R. Chandramouli. Proposed nist standard for role-based access control. *ACM Trans. Inf. Syst. Secur.*, 4:224–274, August 2001.
8. P. W. Fong. Relationship-based access control: protection model and policy language. In *Proc. of the first ACM conference on Data and application security and privacy*, pages 191–202. ACM, 2011.
9. A. Gupta, M. S. Kirkpatrick, and E. Bertino. A formal proximity model for rbac systems. *Computers & Security*, 2013.
10. M. S. Kirkpatrick, M. L. Damiani, and E. Bertino. Prox-rbac: a proximity-based spatially aware rbac. In *Proc. of the 19th ACM SIGSPATIAL Int. Conf. on Advances in Geographic Information Systems*, 2011.
11. I. Ray, M. Kumar, and L. Yu. Lrbac: a location-aware role-based access control model. In *Information Systems Security*, pages 147–161. Springer, 2006.
12. Q. M. S. Osborn, R. Sandhu. Configuring role-based access control to enforce mandatory and discretionary access control policies. In *ACM Transaction on Information and System Security*, 2000.
13. R. Sandhu. Role activation hierarchies. In *In Proceedings of 3rd ACM Workshop on Role-Based Access Control*, 1998.
14. E. Tarameshloo P. Fong. Access control models for geo-social computing systems. In *SACMAT*, 2014.
15. R. K. Thomas. Team-based access control (tmac): a primitive for applying role-based access controls in collaborative environments. In *Proc. of the 2nd ACM workshop on Role-based access control*, 1997.
16. M. Toahchoodee, I. Ray, and R. M. McConnell. Using graph theory to represent a spatio-temporal role-based access control model. *Int. Journal of Next-Generation Computing*, 2010.