Risk Analysis in Access Control Systems Based on Trust Theories

J. Ma†, L. Logrippo†, K. Adi†, S. Mankovski‡,

† *Department of Computer Science and Engineering*
Université du Québec en Outaouais
Québec, Canada.
‡ *CA Labs*
125 Commerce Valley DR W, Thornhill ON, Canada
Email: {ji.ma,luigi,kamel.adi}@uqo.ca, Serge.Mankovskii@ca.com

***Abstract*—There is a need for research on the scientific base and engineering requirements for building trustworthy systems in dynamic environments. To address this need, we study risk analysis for access control from the viewpoint of trust and demonstrate how to extend access control architectures to incorporate trust-based reasoning. We present a theoretical model which allows to reason about and manage risk for access control systems. We also propose a formal approach for establishing and managing theories of trust. The approach can be used for assessing risk and decision making.**

Keywords: Risk analysis, access control system, trust theory, trust degree, risk degree.

I. Introduction

Access control systems are entrusted with the task of determining whether access should be granted for specific requests to access data or other resources. Normally this decision is taken with consideration of risks involved. It is often considered risky to allow data access to untrusted parties, and so access may be denied to them. Related research has been done on Fault Tree Analysis (FTA) [10], [3], Event Tree analysis (ETA) [7], Probabilistic Risk Assessment (PRA) [9], Failure Mode and Effects Analysis (FMEA) [6].

Access control policies are often based on the binary-valued trust model, which has only two different trust degrees, trust (1) and distrust (0). The advantage of using this model is that it is easy to assign or compute trust degrees for agents within a system, theory revisions also can be easily handled. However, in many applications, we may need to consider other models, where the trust degree can be any value between 0 and 1. For example, if the risk degree for accessing a resource is 0.2, the system may consider the access safe; if the risk degree is 0.3, the system may consider it risky. Risk degree may be based on many factors, such as trust, assurance, cost, etc. This requires applying methods for evaluating trust degrees.

Trust is the outcome of observations of agents, and it changes dynamically. When agents lose their trust or gain new trust in a dynamic environment, the theory established based on the initial trust of agents in the system must be revised, otherwise it may no longer be valid [8]. However, there are few papers that discuss the dynamics of trust. Dimmock *et al.* [4] discussed how to extend existing access control architectures to incorporate trust-based evaluation and reasoning. Bhargava *et al.* [2] proposed an approach enhancing role-based access control with trust ratings. Asnar *et al.* [1] proposed an approach to assess risk on the basis of trust relations among actors. This paper is innovative with respect to previous research because it proposes to infer risk from trust.

In this paper, we propose a model for risk analysis in access control mechanisms with consideration of trust. We also show how our approach can be extended for establishing and managing theories.

The rest of this paper is organized as follows. Section 2 presents a trust model for access control systems. Section 3 discusses risk analysis based of theories of trust. Section 4 concludes this paper and discusses further works.

II. Trust Model for Access Control Systems

The notion of trust is fundamental for understanding the interactions between agents such as human beings, machines, organizations, and other entities. In this section we propose a trust model for role based access control systems. Our trust model is defined as:

Definition 1 (Trust Model)**.** *A trust model for access control systems is a 6-tuple.*

$$\mathcal{M} = \langle \mathcal{U}, \mathcal{R}, \mathcal{A}, \mathcal{O}, \mathcal{P}, \mathcal{AR} \rangle.$$

Where $\mathcal{U},\ \mathcal{R},\ \mathcal{A},\ \mathcal{O},\ \mathcal{AR}$ are sets:

- $\mathcal{U}$: a set of users or subjects.
- $\mathcal{R}$: a set of roles.
- $\mathcal{A}$: a set of actions (access, modify, etc.).
- $\mathcal{O}$: a set of objects (documents, records, services, etc).
- $\mathcal{P}$: a set of permissions. In our model, a permission is defined as a pair consisting of an action and an object.
- $\mathcal{AR}$: a set of assignment relations.

The set of assignment relations, $\mathcal{AR}$, includes the following relations:

- RA: role assignment relation, $RA \subseteq \mathcal{U} \times \mathcal{R}$. A user may hold one or more roles.
- PA: permission relation, $PA \subseteq \mathcal{R} \times \mathcal{P}$. A role may hold one or more permission.

Definition 2 (Trust state)**.** *A trust state (S) for a given system is a formal assignment of the trust model of the system.*

For example, suppose that initially in a financial system, we have:

- $\mathcal{U}$ = $\{bob, lisa, tom\}$,
- $\mathcal{R}$ = $\{manager, admin, clerk\}$,
- $\mathcal{A}$ = $\{access, modify, approve\}$,
- $\mathcal{O}$ = $\{loan, record\}$,
- $\mathcal{P}$ = $\{(modify, record), (approve, loan), ...\}$,
- $\mathcal{AR}$ = $\{RA, PA\}$, where

$$RA = \left\{ \begin{array}{l} (bob, manager), \\ (lisa, admin), \\ (tom, clerk) \end{array} \right\}$$

$$PA = \left\{ \begin{array}{l} (manager, (access, record)\ ), \\ (admin, (access, record)\ ), \\ (admin, (modify, record)), \\ (manager, (approve, loan)), \\ ... \end{array} \right\}$$

Definition 3 (Trust theory)**.** *A trust theory for a given system is a formal representation of access control policies and security mechanisms, where security policies are directly transformed to corresponding rules.*

In order to obtain a theory for such systems, we define the following predicates:

- $holds(X, R)$: User $X \in \mathcal{U}$ holds role R, iff $(X, R) \in RA$.
- $has\_permission(R, A, O)$: Role R has the permission to perform action A on object O, iff $(R, (A, O)) \in PA$.
- $user\_permit(X, A, O)$: User X is permitted to perform action A on object O, iff $(X, R) \in RA \wedge (R, (A, O)) \in PA$.
- $is\_user(X)$: X is a user.
- $is\_in(X, Dept)$: User X is in department $Dept$.
- $can\_delegate(X, Y, P)$: User X can delegate to user Y permission $P \in \mathcal{P}$.
- $can\_co\_approve(X, Y, P)$: User X and user Y can co-approve permission $P \in \mathcal{P}$.

The first two predicates correspond to the relations RA and PA in the model, respectively. The other predicates are needed for formalising access control policies.
Suppose that we have the following facts:
F1. $holds(bob, manager)$.
F2. $holds(lisa, admin)$.
F3. $holds(tom, clerk)$.
F4. $has\_permission(manager, access, record)$.
F5. $has\_permission(admin, access, record)$.
F6. $has\_permission(admin, modify, record)$.
F7. $has\_permission(manager, approve, loan)$.

We denote the set of axioms by $\mathcal{F}$:

$$\mathcal{F} = \{F1, F2, F3, F4, F5, F6, F7\}.$$

Now, we consider the following access control policies.

Policy 1 (Permission)**.** *A user can be granted a permission, if he holds an appropriate role.*

The permission policy can be formalised as:
R1. $holds(X, R) \ \wedge has\_permission(R, A, O) \rightarrow user\_permit(X, A, O)$.

Policy 2 (Delegation)**.** *A user can delegate an authorization to another user.*

The delegation policy can be formalised as:
R2. $is\_user(X) \wedge is\_user(Y) \wedge user\_permit(X, A, O) \rightarrow can\_delegate(X, Y, (A, O))$.

Policy 3 (Co-approval)**.** *A permission must be approved by people belonging to two different departments.*

The Co-approval policy can be formalised as:
R3. $is\_in(X, Dept_1) \wedge is\_in(Y, Dept_2) \rightarrow can\_co\_approve(X, Y, P)$.
Here $Dept_1 \neq Dept_2$.

Now, we have established a theory, which includes the fact set $\mathcal{F}$ and three rules.

$$T = \{R1, R2, R3\} \cup \mathcal{F}.$$

The theory provides a foundation for reasoning about the security properties of the system. For example, based on the theory, we can prove that "Lisa is permitted to modify the records". The logical proof outline is given as follows:

Example 1 (Decision Deduction)**.**
(1) $holds(lisa, admin)$. *(F2)*
(2) $has\_permission(admin, modify, record)$. *(F6)*
From (1), (2) and rule R1, we deduce:
(3) $user\_permit(lisa, modify, record)$.

III. Risk Analysis Based on Trust Theories

In this section, we give two examples of analysing risk based on trust theory. We consider two risk scenarios: $delegation$, and $co\text{-}approval$.

Scenario 1 (Delegation Risk)**.** *The company allows managers to approve loans (fact F7), and managers can delegate the authorization to another user (policy R2). However, it may be risky for a manager to delegate loan approval to certain employees, such as an inexperienced employee.*

To convey the notion of risk, we define the following predicates:

- $tv_1(R, P, t)$: The trust degree of role R having permission P is t, where $0 \leq t \leq 1$.
- $delegate(X, Y, P)$: User X delegates to user Y permission P.
- $delegate\_risk(X, Y, P, r)$: The risk degree of user X delegating permission P to user Y is r, where $0 \leq r \leq 1$.
- $delegate\_is\_valid(X, Y, P)$: It is valid for user X to delegate permission P to user Y.

Policy 2 can be replaced by the following two rules:

R2$_1$. $tv_1(R_1, P, t_1) \wedge tv_1(R_2, P, t_2) \wedge holds(X, R_1) \wedge holds(Y, R_2) \wedge delegate(X, Y, P) \rightarrow delegate\_risk(X, Y, P, rv(t_1, t_2))$.

R2$_2$. $delegate\_risk(X, Y, P, r) \wedge (r < v(\varepsilon)) \rightarrow delegate\_is\_valid(X, Y, P)$.

The original delegation policy has been divided into two rules: the first one is used to obtain the risk degree r ($0 \leq r \leq 1$), the second one is the decision making rule, where v is the risk threshold which can be changed. In rule R2$_1$, the risk degree can be calculated by the following function:

$$rv(t_1, t_2) = \begin{cases} 0, \text{when } t_2 \geq t_1 \\ t_1 - t_2, \text{otherwise} \end{cases}$$

Generally, the function used to compute a risk threshold (v) takes its parameters from the environment ε and can be defined in different ways. For instance,

$$v(\varepsilon) = f(x_1 * w_1, ..., x_n * w_2),$$

where x_1, ..., x_n are the factors of trust or risk valuations, respectively, w_1, ..., w_n are associated weights of those factors.

The intuitive idea is: the risk degree of the delegation of X to Y for performing a permission is related to the difference between X′s trust degree and Y′s trust degree. Therefore, if X and Y have the same trust degree for performing a permission, then there is no risk for the delegation. Other formulas are of course possible.

Since we have revised policy 2, we remove rule R2 from the theory, and add rule R2$_1$ and R2$_2$ into it. We then obtain a new theory:

$$T = \{R1, R2_1, R2_2, R3\} \cup \mathcal{F}.$$

Suppose that we have the following facts:
F8. $tv_1(manager, loan\_approval, 1)$.
F9. $tv_1(admin, loan\_approval, 0.5)$.
F10. $tv_1(trainee, loan\_approval, 0)$.
F11. $tv_1(manager, purchase, 1)$.
F12. $tv_1(admin, purchase, 1)$.
F13. $tv_1(trainee, purchase, 0.2)$.
F14. $v(\varepsilon) = 0.3$.

Fact F8 means that the trust degree for managers approving loans is 1. Fact F9 means that the trust degree for administrators approving loans is 0.5, and so on. F14 means the trust threshold of loan approval delegation is 0.3.

Example 2 (Delegation permission with risk consideration)**.**

1) $delegate\_risk(manager, admin, purchase, 0)$.
   In this case, the risk degrees is 0, because the delegatee himself has such permission. Here r = 1 - 1 = 0, the delegation is valid.
2) $delegate\_risk(manager, trainee, loan\_approval, 1)$.
   In this case, the risk degree is higher than the risk threshold (0.3). Here r = 1 - 0 = 1, the delegation is not valid.

Scenario 2 (Co-approval Risk)**.** *According to rule R3, any contract must be approved by people belonging to two different departments. Suppose that the company has an employee, Mary, who belongs to two different departments. So Mary may be able to approve such transactions all by herself, that could be risky.*

For co-approval rules, we further define the following predicates:

- $tv_2(X, Dept, t_i)$: The trust degree of user X in department $Dept$ is t_i.
- $co\_approve(X, Y, P)$: User X and user Y co-approve permission P.
- $co\_approve\_risk(X, Y, P, r)$: The risk degree of user X and user Y co-approving permission P is r.
- $co\_approval\_is\_valid(X, Y, P)$: It is valid for user X and user Y to co-approve permission P.

Policy 3 can be replaced by the following two rules:

R3$_1$. $tv_2(X, Dept_1, t_x) \wedge tv_2(Y, Dept_2, t_y) \wedge co\_approve(X, Y, P) \rightarrow co\_approve\_risk(X, Y, P, rv_1(t_x, t_y))$.

R3$_2$. $co\_approve\_risk(X, Y, P, r) \wedge (r < v(\varepsilon)) \rightarrow co\_approval\_is\_valid(X, Y, P)$.

The original policy has been divided into two rules: the first one is used to obtain the risk degree r ($0 \leq r \leq 1$), the second one is the decision making rule, where v is the risk threshold which can be changed. In rule R3$_1$, the risk degree is calculated by the following function:

$$rv_1(t_x, t_y) = 1 - t_x * t_y$$

Since we have revised policy 3, we remove rule R3 from the theory, and add rule R3$_1$ and R3$_2$ into the theory. We then obtain a new theory:

$$T = \{R1, R2_1, R2_2, R3_1, R3_2\} \cup \mathcal{F}.$$

Suppose that we have the following facts:
F15. $tv_2(mary, dept1, 0.5)$.
F16. $tv_2(mary, dept2, 0.5)$.
F17. $tv_2(bob, dept1, 1)$.

F18. $tv_2(peter, dept1, 1)$.
F19. $tv_2(john, dept2, 1)$.
F20. $dept_1 \neq dept_2$.
F21. $v(\varepsilon) = 0.2$.

If a person X is in department $dept1$, we write $tv_2(X, dept1, 1)$, if X is not in department $dept1$, we write $tv_2(X, dept1, 0)$. Since Mary belongs to two departments, then $tv_2(mary, dept1, 0.5)$ and $tv_2(mary, dept2, 0.5)$. The meaning of Axioms F17 - F20 is obvious. F21 means the trust threshold of contract Co-approval is 0.2.

Example 3 (Co-approval permission with risk consideration)**.**

1) $co\_approve\_risk(bob, john, P, 0)$.
   Bob is in department $dept1$, John is in department $dept2$. In this case, there is no risk. Here r = 1 - 1 * 1 = 0.
2) $co\_approve\_risk(mary, mary, P, 0.75)$.
   Mary represents both departments $dept1$ and $dept2$. In this case, the risk degree is higher than the risk threshold (0.2). Here r = 1 - 0.5 * 0.5 = 0.75.
3) $co\_approve\_risk(bob, mary, P, 0.5)$.
   Bob is in department $dept1$, Mary represents department $dept2$. In this case, the risk degree is higher than the risk threshold (0.2). Here r = 1 - 1 * 0.5 = 0.5.

The above examples have illustrated a method for performing risk analysis in access control systems with the notion of trust. This procedure involves the following steps:

1) Building a trust model for a given system. *(Def. 1)*
2) Defining appropriate predicates used to express trust and risks.
3) Formalising policies, that is defining rules for decision making. These rules form a theory of trust for the system. *(Def. 3)*
4) Revising rules based on identified risk scenarios.

A prototype of the system proposed above was implemented in Prolog. Due to the limitation of space, we only list the following rules:

R2$_1$. $delegate\_risk(A, B, Task, R) :-$
$delegate(A, B, Task), tv(A, Task, Ta), tv(B, Task, Tb),$
$R = Ta - Tb$.
R2$_2$. $delegate\_approval(A, B, Task) :-$
$delegate\_risk(A, B, Task, R), R < 0.3$.
R3$_1$. $co\_approve\_risk(A, B, contract, R) :-$
$tv(A, P, Va), tv(B, Q, Vb),$
$is\_different(P, Q), co\_approve(A, B, contract),$
$R = 1 - Va * Vb$.
R3$_2$. $co\_approve\_is\_valid(A, B, contract) :-$
$co\_approve\_risk(A, B, contract, R), R < 0.2$.

IV. Conclusion

We have proposed a formal approach for establishing and managing theories of trust for risk analysis in access control systems. There are no existing general and systematic techniques or tools for risk analysis in such systems. Therefore the methods and techniques proposed in this paper have potential for many diverse applications.

Risk degree is based on many factors, such as trust, assurance, cost, etc. Therefore, for risk management, the following issues should be investigated: how to evaluate risk degrees, and how to determine the trust threshold for a given system. Different trust thresholds may lead to different policy implementation.

There are several methods and techniques for belief revision that could be helpful for theory revision. We plan to investigate a variety of belief revision techniques that can be applied for the revision of trust theories. The controlled revision approach of Gabbay et al. [5] may be particularly useful for practical applications.

Acknowledgement

This research has been funded in part by grants from PROMPT Québec and from CA Labs.

References

[1] Y. Asnar, P. Giorgini, F. Massacci, and N. Zannone. From trust to dependability through risk analysis. In *ARES*, pages 19–26, 2007.

[2] B. K. Bhargava and L. Lilien. Vulnerabilities and threats in distributed systems. In *ICDCIT*, pages 146–157, 2004.

[3] J. Dehlinger and J. B. Dugan. Dynamic event/fault tree analysis of multi-agent systems using galileo. In *QSIC*, pages 429–434, 2008.

[4] N. Dimmock, A. Belokosztolszki, D. M. Eyers, J. Bacon, and K. Moody. Using trust and risk in role-based access control policies. In *SACMAT*, pages 156–162, 2004.

[5] D. Gabbay, G. Pigozzi, and J. Woods. Controlled revision - an algorithmic approach for belief revision. *Journal of Logic and Computation*, 13(1):3–22, 2003.

[6] L. Grunske, R. Colvin, and K. Winter. Probabilistic model-checking support for FMEA. In *QEST*, pages 119–128, 2007.

[7] D. Huang, T. Chen, and M. J. Wang. A fuzzy set approach for event tree analysis. *Fuzzy Sets and Systems*, 118(1):153–165, 2001.

[8] J. Ma and M. A. Orgun. Trust management and trust theory revision. *IEEE Transactions on Systems, Man and Cybernetics, Part A*, 36(3):451–460, 2006.

[9] H. Nejad, D. Zhu, and A. Mosleh. Hierarchical planning and multi-level scheduling for simulation-based probabilistic risk assessment. In *Winter Simulation Conference*, pages 1189–1197, 2007.

[10] H. Sun, M. Hauptman, and R. R. Lutz. Integrating product-line fault tree analysis into aadl models. In *HASE*, pages 15–22, 2007.