Insider threat likelihood assessment for access control systems: Quantitative approach

Sofiene Boulares, Kamel Adi, and Luigi Logrippo

Département d'informatique et d'ingénierie
Université du Québec en Outaouais
Gatineau, QC, Canada
{bous42,kamel.adi,luigi.logrippo}@uqo.ca

Abstract. Organizations need to use flexible access control mechanisms where the access decisions to critical information assets are taken dynamically. In this paper, we present a framework for insider threat likelihood assessment within the context of access control systems. Our approach takes into account information flows, the trustworthiness of subjects, the sensitivity of objects and the security countermeasures. We identify and formally describe a set of properties to be satisfied within this approach. These properties are, then used for quantitatively assessing the insider threat likelihood.

Keywords: Information Security, Access control, Information flow, Insider threat, Threat likelihood assessment, Risk assessment.

1 Introduction

Risk-based access control provides support for flexible access control decisions and facilitates information sharing. Consider a situation where a workflow architect asks an IT security specialist to determine which combinations of operations are less risky for the tasks composing a workflow, given the subjects, objects and actions involved in each operation. The decisions could be based on the evaluation of access risks, by selecting the combinations giving the lowest risk values.

An access control system that can give employees risky accesses can cause insider security incidents. According to the US firm Forrester Research, insider incidents within organizations represent 46% of security breaches [11]. In addition, the survey Global Corporate IT Security Risks 2013 [6], conducted by Kaspersky Lab, shows that 85% of companies worldwide have experienced an insider computer security incident.

Bishop et al [3] distinguish two categories of insider threats:

1. violation of access control policy by using authorized access,
2. violation of security policy by obtaining unauthorized access.

Our approach for threat likelihood estimation of access requests deals with the first category of insider threats which includes cases where an employee uses his legitimate access to perform an action that violates the access control policy: discloses sensitive data to a third party, releases information to untrusted environments, etc. Our method can be seen as an approach to estimate the threat likelihood of the violation of an access control policy, caused by the authorization of other access requests.

The rest of the paper is organized as follows. Section 2 presents an overview of our work and the contribution of this paper. In Section 3, we present our threat assessment approach. In Section 4, we compare our work with notable work of the literature and we present the limitations of our approach. Finally, we draw conclusions for this paper and outline opportunities for future work in Section 5.

2 Overview and contribution

Assessing the threat likelihood for different types of events with their predicted impacts is a common way to assess IT risks. OWASP [9] defines the risk R as "the product of the likelihood L of a security incident occurring times the impact I that will be incurred by the organization due to the incident, that is: $R = L \times I$".

Our approach differentiates between the *intrinsic threat likelihood* which is the probability that the risk in question will occur in the absence of security countermeasures and *threat likelihood* which considers the reduction of risk by the application of countermeasures [5]. The security countermeasures could be devices, procedures, or techniques that reduce the likelihood of threat on the security of information that is processed, stored or transmitted. Examples of such countermeasures are enabled access logs, data encryption, etc.

Let us assume the existence of the following entities: S a set of subjects, O a set of objects, A a set of actions, L_c a set of secrecy levels, and SC a set of security criteria. We limit the set A to two actions, read and write, which will be collectively called *accesses*. We also limit the set SC to two criteria: Secrecy and Integrity. We define a function $Threat\_likelihood : S \times A \times O \times SC \rightarrow [0, 1]$ that represents the threat likelihood value when a subject $s \in S$ requesting an action $a \in A$ on an object $o \in O$ when a security criterion $sc \in SC$ is intended. Secrecy will be abbreviated c.

3 Assessment of threat likelihood when secrecy is intended

In this section, we propose our approach to estimatethreat likelihoodon secrecy in access control systems. This approach considers the following factors: the intended security criteria (secrecy in this section), the requested action (read or write), the secrecy level of subjects requesting access, the secrecy level of objects to be accessed and the security countermeasures. We assume that threat likelihood depends on the importance of information flow between objects and subjects, determined by the difference between their security levels.

In our approach, the likelihood of threat on secrecy increases when information flows down. Consider, for example, the information flow when a *Top Secret* subject writes in a *Public* object, such information flow is more important than the one when the same subject writes in a *Secret* object. In the first case, *Top Secret* information could be leaked to the public, in the second case this information would remain secret. It is reasonable to assume that the threat likelihood would be higher in the first case. The reasoning for integrity is dual.

We define a total order on L_c and for each secrecy level in L_c, we assign a numerical value in accordance with the defined order, where higher numbers denote higher security levels. Throughout this paper, the following functions will be needed to develop our approach:

- $csl : S \rightarrow L_c$ formally represents the assignment of secrecy levels to subjects that reflects the trust bestowed upon each of them.
- $col : O \rightarrow L_c$ formally represents the assignment of secrecy levels to objects that reflects the protection needs of the data.

3.1 Defining "threat likelihood"

Instead of adopting the binary vision of the *Bell La Padula* model [2] to assess the threat likelihood of read and write requests, we propose the following principles: we consider that permitting a subject s to read an object o, such that $csl(s) < col(o)$ or permitting a subject s to write in an object o, such that $csl(s) > col(o)$, presents by itself a measurable threat likelihood.

In this section, we define the "threat likelihood" on secrecy as follows: we say that the likelihood of threat on secrecy is non null if a subject $s \in S$ is able to read an object $o \in O$, such that $csl(s) < col(o)$. But for any attempt by a subject s to read an object o, such that $csl(s) \geq col(o)$ the threat likelihood is null. Any measure of read threat likelihood on secrecy in the first case is affected by the following two general principles:

- **Principle 1:** the likelihood of threat on secrecy increases (or decreases) as the object's secrecy level increases (respectively decreases).
- **Principle 2:** the likelihood of threat on secrecy increases (or decreases) as the subject's secrecy level decreases (respectively increases).

The reasoning for write accesses is dual.

We define the relation $<_T$ in the following way: $(s, a, o, sc) <_T (s', a', o', sc)$ iff $Threat\_likelihood(s, a, o, sc) < Threat\_likelihood(s', a', o', sc)$.

3.2 Read threat likelihood assessment for secrecy

We assume the existence of the subjects: s_1, s_2, s_3, s_4, s_5 and s_6, and the objects o_1 and o_2. Table 1(a) and Table 1(b) illustrate the secrecy levels of these entities.

Subjects	s_1	s_2	s_3	s_4	s_5	s_6
Secrecy levels	4	3	2	1	1	1

(a)

Objects	o_1	o_2
Secrecy levels	5	4

(b)

Table 1. Secrecy levels for running examples.

3.2.1 Read threat likelihood assessment for secrecy: qualitative approach

Assume that access for data objects has been requested by subjects who are employees of the business that owns the objects (trusted and reliable to some degree by the system). In this case, data owners might be more concerned about the secrecy levels of objects than the secrecy levels of subjects. Hence, our approach for threat likelihood assessment in this paper is primarily based on the *secrecy levels of objects.*

Let us assume that a workflow architect asked an IT security specialist to define a set of tasks composing a workflow by selecting the least threatening combinations of subjects, objects and actions for the secrecy of data. Task T_1 can be executed by s_2 reading from objects o_1 or o_2, task T_2 can be executed by either s_3 or s_4 reading from o_2 and task T_3 can be executed by either s_5 or s_6 reading from o_1. The last two subjects request access from two distant sites where s_5 is connected via an unencrypted public network and s_6 via VPN.

To determine the least threatening combinations of subjects, objects and actions on secrecy we follow this method:

Method 1: A read threat likelihood assessment technique that is primarily based on object secrecy levels should support the following:

1. always apply **Principle 1**: read threat likelihood always increases as object secrecy level increases,
2. whenever object secrecy levels are the same, apply **Principle 2**: read threat likelihood increases as subject secrecy level decreases,
3. apply **Principle 3**: threat likelihood of accesses increases (or decreases) as the effect of security countermeasures reducing the threat likelihood decreases (respectively increases).

The least threatening combinations of our example according to **Method 1** are as follows: T_1 should be executed by s_2 reading from o_2 since $col(o_2) < col(o_1)$ $((s_2, r, o_2, c) <_T (s_2, r, o_1, c))$, T_2 should be executed by s_3 reading from o_2 since $csl(s_3) > csl(s_4)$ $((s_3, r, o_2, c) <_T (s_4, r, o_2, c))$ and T_3 should be executed by s_6 reading from o_1 since $csl(s_5) = csl(s_6)$ and only s_6 is connected via VPN which is a countermeasure that reduces threat likelihood by preventing disclosure of information $((s_6, r, o_1, c) <_T (s_5, r, o_1, c))$. Indeed, VPNs typically allow remote access using tunnelling protocols and encryption techniques.

3.2.2 Read threat likelihood assessment for secrecy: quantitative approach

Let us consider task T_4 that can be executed by either s_1 or s_2 reading from

o_1 where s_1 is connected via an unencrypted public network and s_2 via VPN. According to **Principles 1** and **2**, allowing s_2 to read object o_1 has a greater likelihood of threat on secrecy than allowing s_1 to read object o_1. However, **Principle 3** tells us that this may not be true in the presence of countermeasures such as the VPN, that can reduce the threat likelihood of s_2 reading o_1. Hence, we can see that priority orders such as the one outlined in section 3.2.1, can not permit threat likelihood comparison in all cases. However, quantitative measures which correspond to this threat likelihood ordering may be useful, such as in the case of task T_4. There can be many different formulas which respect the properties of our approach and can measure the threat likelihood of granting access. In this section, we propose a formula and describe its construction.

ISO / IEC 27001 [10] requires regular verification of computer security. In order to determine to which extent the countermeasures are producing the desired outcome to meet the security requirements, the security administrator measures the contribution of the implemented countermeasures in the reduction of risks. In this work, we consider the effect of countermeasures in the calculation of threat likelihood. In table 2, each rule determines a countermeasure and its effect corresponding to an access request identified by the subject's security level, the object's security level, the action requested and the security criteria intended.

Table 2 shows a representation of all possible read accesses by subjects to objects when secrecy is intended. Note that when $csl(s) > col(o)$, the threat likelihood is null. Hence, entries of Table 2 are empty along or below the diagonal. Otherwise, each table entry $[i, j]$ includes a set of couples (measure, value) that represents the countermeasures and their contribution in the reduction of threat likelihood of a subject s reading an object o, where $csl(s) = i$ and $col(o) = j$. The sum of all countermeasures values in each entry is bound between 0 and 1.

The rule of entry [2, 4] shows that if a subject having a secrecy level 2 reads an object having a secrecy level 4, then the countermeasures m_3 and m_4 can respectively reduce the likelihood of threat on secrecy by 0.5 and 0.2.

$Counter(s, a, o, sc)$ denotes the sum of the effects of the different implemented countermeasures to reduce threat likelihood if s executes an action a on an object o when the security criteria sc is intended. For example, we can see from Table 2 that if a subject s having a secrecy level of 1 requests to read an object o having a secrecy level of 5 when secrecy is intended and all three countermeasures are applied, we have $Counter(s, r, o, c) = 0.5 + 0.2 + 0.2 = 0.9$.

We define the following additional principles for the calculation of the threat likelihood of access requests, which we assume to be bound between 0 and 1.

- **Principle 4:** The threat likelihood of an access request is equal to zero, if the cumulative effect of the corresponding security countermeasures is equal to or greater than the value of the intrinsic threat likelihood.
- **Principle 5:** The threat likelihood of an access request increases (or decreases) when the intrinsic threat likelihood increases (respectively decreases).

We now introduce the concept of threat likelihood indexing. We associate a numerical value representing the threat likelihood index from the set $\{0, \cdots, |L_c| - 1\}$ to each subject and object having a secrecy level in L_c. In the case of read

Subjects secrecy levels	Objects secrecy level 1	Objects secrecy level 2	Objects secrecy level 3	Objects secrecy level 4	Objects secrecy level 5
1		$(m_5, 0.5)$	$(m_5, 0.5)$	$(m_3, 0.5)$	$(m_1, 0.5)$ $(m_2, 0.2)$ $(m_4, 0.2)$
2			$(m_2, 0.2)$ $(m_4, 0.2)$	$(m_3, 0.5)$ $(m_4, 0.2)$	$(m_2, 0.2)$ $(m_4, 0.2)$
3				$(m_3, 0.5)$	$(m_4, 0.2)$
4					$(m_4, 0.2)$
5					

Table 2. The effect of countermeasures in the reduction of the read threat likelihood.

accesses when secrecy is intended, from the point of view of subjects, we expect the threat likelihood to increase as subject secrecy levels decrease. Hence, subject threat likelihood index values decrease with subject secrecy levels. For *level* in L_c, we write $\widehat{level}$ to denote a subject threat likelihood index. Formally, $(\widehat{level})$ = $|L_c| - level$. For example, when L_c = {Top secret, Secret, Confidential, Restricted, Public}, $(\widehat{Secret})$ = 5 - 4 = 1. However, object threat likelihood indexes increase with object secrecy levels. We write $\overbrace{level}$ to denote an object threat likelihood index. Formally, $\overbrace{level} = level$ -1. For example, $\overbrace{Secret} = 4 - 1 = 3$.

If we assume that $|L_c| = 5$ there can be at most 5 × 5 = 25 combinations of subject-object accesses. We define a function $Intrinsic : S \times A \times O \times SC \rightarrow [0, 1]$ that represents the intrinsic threat likelihood value of a subject $s \in S$ requesting an action $a \in A$ on an object $o \in O$ when a security criterion $sc \in SC$ is intended.

$$Intrinsic(s, r, o, c) = \begin{cases} \frac{(|L_c| \times \overbrace{col(o)} + \widehat{csl(s)})}{(|L_c|^2) - 1}, \textbf{if } csl(s) < col(o) \\ 0, \quad \textbf{Otherwise}. \end{cases} \tag{1}$$

A formula that respects the principles of **Method 1** and **Principles 4** and **5** for measuring the threat on secrecy likelihood of granting read access to a subject s for an object o, is given below:

$$Threat\_likelihood(s, r, o, c) = \begin{cases} Intrinsic(s, r, o, c) - Counter(s, r, o, c), \\ \textbf{if } csl(s) < col(o) \ and \\ Counter(s, r, o, c) < Intrinsic(s, r, o, c) \\ 0, \quad \textbf{Otherwise}. \end{cases} \tag{2}$$

The numerator of formula (1) is intuitive. Since we require that more importance be given to the threat likelihood index of objects, we multiply the object threat likelihood index by $|L_c|$ that equals the cardinality of the set of secrecy levels L_c. Then, we add the threat likelihood index of the subject. The numerator of the formula maps all possible read accesses by subjects to objects into an interval $[0 \cdots (|L_c|^2) - 1]$, where a higher value represents a greater threat likelihood. In order to have intrinsic likelihood threat values into an interval $[0, 1]$, we divide the value obtained from the numerator by $(|L_c|^2)$ - 1. In formula (2), we subtract the value representing the effect of the different implemented countermeasures corresponding to the request in question. The resultant value represents the object-based read threat likelihood that respects the principles of **Method 1** and **Principles 4** and **5**.

Let us consider that the coutermeasure m_2 in Table 2 represents the encryption of data and we apply formula (2) to our example stated in 3.2.2. We have $Counter(s_1, r, o_1, c) = 0$ and $Counter(s_2, r, o_1, c) = 0.2$. We get the following: $Threat\_likelihood(s_1, r, o_1, c)$ = $Intrinsic(s_1, r, o_1, c)$ - $Counter(s_1, r, o_1, c)$ = 0.87 (1) and $Threat\_likelihood(s_2, r, o_1, c) = Intrinsic(s_2, r, o_1, c)$ - $Counter(s_2, r, o_1, c)$ = 0.91 - 0.2 = 0.71 (2). From (1) and (2), we have $Threat\_likelihood(s_2, r, o_1, c) < Threat\_likelihood(s_1, r, o_1, c)$.

Future papers will show how to derive formulas giving values representing the object_based likelihood of threat on secrecy when write access is requested and the likelihood of threat on integrity when write and read accesses are requested. Note that threat likelihood on integrity increases when information flows up.

4 Related work and limitations

Cheng et al propose Fuzzy Multi-Level Security (Fuzzy MLS), which quantifies the risk of an access request in multi-level security systems as a product of the value of information and probability of unauthorized disclosure [4]. Unlike Fuzzy MLS which is limited to the estimation of the threat likelihood of read accesses forbidden by Bell La Padula, our approach estimates the threat likelihood of read and write accesses, is applicable when the objective of integrity is of interest (is not limited to secrecy) and considers security countermeasures mitigating the threat likelihood.

Bartsch proposes a policy override calculus for qualitative risk assessment in the context of role-based access control systems [1]. This work presents a qualitative estimation of threat likelihood. In comparison with the work of Bartsch, our approach is both qualitative and quantitative, developed in the context of generic access control systems and is not limited to RBAC.

Threat likelihood assessment in our framework cannot cover unexpected threats such as those in which several other socio-technical parameters must be taken into consideration for reflecting the reality of insider threats such as users' access history, behavior, collusion with other users, etc.

5 Conclusion

The main contribution of this paper is a quantitative approach for threat likelihood assessment in the context of access control systems. Our approach considers primarily the security levels of objects, and thus gives more priority to the sensitivity of data. This is only one possibility and our approach can be easily modified to accommodate other views, such as those presented in [8, 7]. In order to be compliant with IT Risk standards and guidelines, and to obtain realistic values of threat likelihood, our approach takes account of the effect of the security countermeasures mitigating the threat likelihood of access requests.

In this paper, we have focused on quantitative threat likelihood assessment, which is a pre-requisite for estimating access risks. However, our ultimate goal is to develop a framework for estimating the risk of access requests.

Acknowledgements. This research was partially supported by the Natural Sciences and Engineering Research Council of Canada.

References

1. Steffen Bartsch. A calculus for the qualitative risk assessment of policy override authorization. In Proceedings of the 3rd international conference on Security of information and networks, pages 62–70. ACM, 2010.
2. D Elliott Bell and Leonard J La Padula. Secure computer system: Unified exposition and multics interpretation. Technical report, DTIC Document, 1976.
3. Matt Bishop and Carrie Gates. Defining the insider threat. In Proceedings of the 4th annual workshop on Cyber security and information intelligence research, page 15. ACM, 2008.
4. Pau-Chen Cheng, Pankaj Rohatgi, Claudia Keser, Paul A Karger, Grant M Wagner, and Angela Schuett Reninger. Fuzzy multi-level security: An experiment on quantified risk-adaptive access control. In 2007 IEEE Symposium on Security and Privacy (SP'07), pages 222–230. IEEE, 2007.
5. Clusif. MEHARI 2010 principes fondamentaux et spécifications fonctionnelles. Club de la sécurité de l'information français, 2009.
6. IT Global Corporate. Security risks (2013).
7. Hemanth Khambhammettu, Sofiene Boulares, Kamel Adi, and Luigi Logrippo. A framework for threat assessment in access control systems. In IFIP International Information Security Conference, pages 187–198. Springer, 2012.
8. Hemanth Khambhammettu, Sofiene Boulares, Kamel Adi, and Luigi Logrippo. A framework for risk assessment in access control systems. Computers & Security, 39:86–103, 2013.
9. M. MEUCCI and A. Muller. The owasp testing guide 4.0, 2014.
10. International organization for Standardization. ISO/IEC 27001: Information Technology, Security Techniques, Information Security Management Systems, Requirements. ISO/IEC, 2005.
11. Heidi Shey, K Mak, S Balaouras, and B Luu. Understand the state of data security and privacy: 2013 to 2014. Forrester Research Inc. October, 1, 2013.