Information flow-based security levels assessment for access control systems

Sofiene Boulares, Kamel Adi, Luigi Logrippo

Département d'informatique et d'ingénierie
Université du Québec en Outaouais
Gatineau, QC, Canada

{bous42, kamel.adi, luigi.logrippo}@uqo.ca

Abstract. Access control systems are designed to allow or deny access to data according to organizational policies. In many organizations, the access rights of subjects to data objects are determined in consideration of clearance levels of subjects and classification levels of objects. In most formally-defined traditional access control systems, levels are predetermined and policies are rigid. However, in practice organizations need to use flexible methods where the levels are determined dynamically by information flow criteria. In this paper, we develop a method that is both formal and flexible to determine entities' security levels on the basis of access history, which characterizes the data that subjects can know or objects can contain. Our approach is motivated with a number of different examples, showing that the method meets real-life organizational requirements.

Keywords: Information security, Access control, Information flow, Access history, Clearance, Classification.

1. Introduction

Organizations are dependent upon information to do business and information requires protection for confidentiality, integrity and availability. For this reason, data objects (files, databases, etc.) are typically marked with classification levels e.g. unclassified, restricted, confidential, secret and top secret, to protect personal privacy or competitive secrets. The choice of objects classifications is often based on impact assessment and access to information is restricted by security policies to particular trusted subjects (process, machine, etc.). Thus, security clearance is required to access classified information.

Security levels (clearances and classifications), such as levels in Multi-Level Systems (MLS) [1] are often assumed to be exact and correct. In reality, levels are assigned empirically and could result in too restrictive or too permissive policies. If levels were more accurate, they could be used to take better access control decisions. This would help reduce the risks to the information and the organization's ability to conduct its missions.

For example, suppose that an object, initially of low classification level, has been allowed to store more highly classified information, then its classification level should increase so that future access control decisions can take into consideration this fact. Situations such as these can occur in many enterprises, and especially in highly dynamic environments such as the Web or the Cloud.

To this end, the main contribution of this paper is to propose a set of precise principles to determine object classifications and subject clearances. Our approach has the potential to address issues related to inference problems (information association and information aggregation).

Information flow is the transfer of information from subjects to objects and vice versa. Some information flows are more important than others, because of their possible consequences. For example, the information flow of a Top secret subject writing on an Unclassified object is more dangerous than the information flow of the same subject writing 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. To our knowledge, only two access control models known in the literature today are based on concepts of classification and information flow: the High-water mark [3] which predates the Bell Lapadula model [4] and the Low-water mark [3] which is an extension of the Biba Model [5]. These models, known under the collective names of Multi-Level Security (MLS) models, will be discussed in Section 5.

The access history of subjects to objects and the resulting possible information flow could have an impact on their security levels. We will see that information can be transferred not only directly, but also by association and aggregation. In our access control model, we consider information that can be obtained or inferred from previous accesses and information that could be inferred from these and the data now requested. Information inference could present a security breach if more highly classified information can be inferred from less classified information [2].

To determine when more highly classified information can be inferred from less classified information, we need to determine precisely what a subject can know and what an object can store.

More precisely, there are two important cases of information inference:

- **Information aggregation,** that occurs whenever there is a collection of data items that is classified at a higher level than the levels of individual data items by themselves [2].
  Example. The content of a single medical file is Secret, but the aggregate information concerning all the medical files is Top Secret.
- **Information association,** that occurs whenever two values seen together are classified at a higher level than the classification of either value individually [2].
  Example. The list consisting of the names of all employees and the list containing all employees' social insurance numbers have low confidentiality levels, while a combined list giving employees names with their social insurance numbers has a high confidentiality level.

Thus, information flow is an important element in order to decide security levels of subjects and objects, and it is determined by the history of subjects accessing objects.

In this paper, we present a history and an information flow-based approach to determine subjects and objects security levels. We use many examples as a basis for developing and identifying a set of principles.

The rest of the paper is organized as follows. Section 2 presents a set of basic concepts for our approach. Section 3 describes our subjects and objects confidentiality levels assessment approach. Section 4 shows how to assess subjects and objects integrity levels. In Section 5, we compare our work with related works of the literature. We draw conclusions and outline opportunities for future work in Section 6.

2. Basic concepts for our approach

We assume the existence of the following entities: S a set of subjects, O a set of objects and L a set of security levels. Members of these sets are denoted by lower-case letters s, o, and l with subscripts and primes. According to [2] and [6], confidentiality is related to disclosure of information, while integrity is related to modification of information. In our approach, confidentiality levels of subjects and objects increase when information can flow down to them, and their integrity levels decrease when information can flow up to them. These ideas are behind the properties of the mentioned MLS models.

We adapt the following concepts presented in [6] as follows:

- Two relationships between subjects and objects or between objects respectively: CanKnow and CanStore.
- Two relationships that express previous accesses between subjects and objects: HasRead and HasWritten.

We use the following abbreviations:
CK for CanKnow, CS for CanStore, HR for HasRead and HW for HasWritten.

Table 1 defines two rules: The rule for CK expresses the fact that, if there exists a subject who has read an object, then the subject can know information from that object. The rule for CS deals with storing and expresses the fact that information transfer can occur between objects by effect of subjects reading from and writing in objects. Throughout this paper, we will use the intuitive meaning of these rules in place of their logic formulation.

Table 1. Deductive system

1. The inference rule for CK is: HR(s,o)→ CK(s,o)(If s has read o, then s can know information from o) 2. The inference rule for CS is: HR(s,o) ∧ HW(s,o'))→ CS(o',o)(If s has read from o and s has written in o',then o' can store information from o) CS(o,o) is always true.

We also define the functions CSS and CKS, as follows:

- For any s, CanKnowSet(s) or CKS(s) is the set of objects o for which CK(s, o) is true: $CKS(s) =_{def} \{o \mid CK(s, o) = true\}$.

- For any o, CanStoreSet(o) or CSS(o) is the set of objects o' for which CS(o, o') is true: $CSS(s) =_{def} \{o' \mid CS(o, o') = true\}$.

The following simple example will introduce the idea of our method. Consider a system with two subjects s_1 and s_2 and two objects o_1 and o_2. Suppose we have the following:

a. $HR(s_1,o_1)$: s_1 has read o_1.
b. $HW(s_1,o_2)$: s_1 has written in o_2.
c. $HR(s_2,o_2)$: s_2 has read o_2.

We can conclude that $CKS(s_1) = \{o_1\}$, $CSS(o_2) = \{o_2, o_1\}$ and $CKS(s_2) = \{o_2, o_1\}$. In other words, subjects can know information by reading them from objects, and objects can store information that is written in them by subjects. Objects can also contain information initially. The effects of the relationships a, b and c, can be shown in figure 1, where subjects are represented by rectangles and objects by circles. A rectangle containing a circle means that the subject can know data from that object. A circle containing a circle means that the object can store data from that object.
Figure 1 (a) shows that s1 can know data from o1.
Figure 1 (b) shows that o2 can store data from o1.
Figure 1 (c) shows that s2 can know data from o1 and o2.

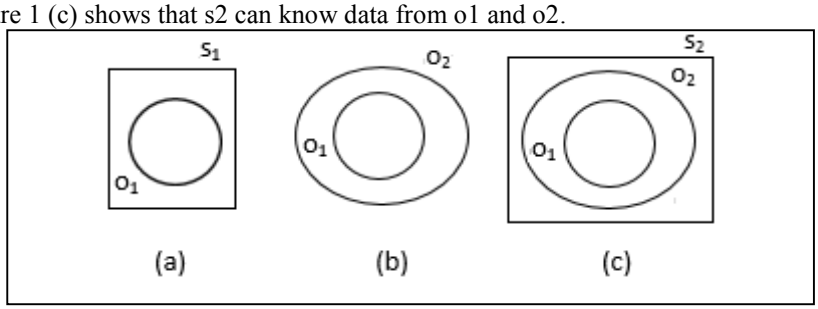

Figure 1. Effects of a, b and c

3. Access history and information flow-based confidentiality levels assessment

Throughout this section, we present a series of examples for developing the conceptual foundation of our history and information flow-based confidentiality levels assessment approach. We consider that confidentiality levels of subjects and objects have been previously assessed at initial values. They can change as a result of information flow: a Read action creates an information flow from an object to a subject and a Write action creates an information flow from a subject to an object. Subjects can increase their confidentiality levels as they acquire information from higher levels and objects can increase their confidentiality levels as they store information from higher levels. The number of accesses is another factor to be considered when assessing subjects and objects confidentiality levels. We define a total order on L and for each security level in L, we assign a numerical value corresponding to the defined order. For example, if L = {Unclassified, Restricted, Classified, Secret, Top Secret}, then the value Unclassified corresponds to the number 1, Restricted corresponds to the number 2 and so on. To simplify the notation, L will be considered to be understood and so it won't

need to be mentioned: in each system, there is only one L which applies to subjects as well as objects. Throughout this paper, the following concepts will be needed to develop our approach:

- sl : S → L formally represents the assignment of security levels to subjects that reflects the trust bestowed upon each of them by the organization that owns the data.
- ol : O → L formally represents the assignment of security levels to objects that reflects the protection needs of the data each of them holds.
- For s ∈ S, CKSL(s) is the multiset of levels of objects o for which CK(s, o) is true, in addition to the level of the subject assigned by the administrator.
  Example. CKSL(s) = {l',l, l'} means that subject s can know data from two different objects belonging to level l' and one object belonging to level l.
- For s ∈ S, $CKSL^{+}$(s) is the multiset of levels of objects o for which CK(s, o) is true such that sl(s) ≤ ol (o).
  Example. $CKSL^{+}$(s) = {l',l''} means that subject s can know data from two different higher level objects belonging to levels l' and l''. So sl(s) ≤ l' and sl(s) ≤ l''.
- For o ∈ O, CSSL(o) is the multiset of levels of objects o' for which CS(o,o') is true.
  Example. CSSL(o) = {l',l ,l'} means that object o can store two different objects belonging to level l' and one object belonging to level l.
- For o ∈ O, $CSSL^{+}$(o) is the multiset of levels of objects o' for which CS(o, o') is true such that ol(o) ≤ ol (o').
  Example. $CSSL^{+}$(o) = {l',l''} means that object o can store data from two different higher level objects belonging to levels l' and l''. So ol(o) ≤ l' and ol(o) ≤ l''.

3.1. Access history and information flow-based subjects' confidentiality levels assessment

As mentioned, a main assumption in this paper is that the confidentiality levels of subjects may be assessed by referring to their read access history. We further justify this assumption with relation to a number of intuitively reasonable requirements or principles, as follows:

Principle 1: The confidentiality level of a subject increases as the subject reads objects having confidentiality levels equal to or greater than its own, i.e. as information it can know grows.

Principle 2: The confidentiality level of a subject increases as the subject reads greater *numbers* of objects having confidentiality levels equal to or greater than its own.

Principle 3: For a subject that has no history of reading objects, the confidentiality level is set to a minimum/default value. This can be determined by the system administrator.

Running example. In this section, we describe a scenario that will be used in the rest of the paper for motivating our approach. Table 2(a) shows the confidentiality levels of

the following subjects: Nadia, Claude, Bruno, Carl and Sabrina. Table 2(b) shows the confidentiality levels of the following objects o_1, o_2, o_3, o_4, o_5, o_6, o_7 and o_8.

The objective of our work in this section, is to compare subjects confidentiality levels. Hence, throughout this section, we only cite examples where $sl(s) \leq ol(o)$ for read accesses because subjects confidentiality levels change only when subjects can know higher information at levels equal to or higher than their own.

Table 2.Confidentiality levels

Subject	Confidentiality level
Nadia	2
Claude	2
Bruno	1
Carl	1
Sabrina	1

(a)

Object	Confidentiality level
o_1	4
o_2	4
o_3	3
o_4	2
o_5	1
o_6	1
o_7	1
o_8	1

(b)

We now give examples that motivate our technique.

Example 1. Suppose a case where Bruno has read object o_1 and Carl has read object o_4. We have the following from Table 2: $sl(Bruno) = 1$, $sl(Carl) = 1$, $ol(o_1) = 4$ and $ol(o_4) = 2$. So Bruno and Carl start at the same confidentiality level. If Bruno reads object o_1 and Carl reads o_4, then according to **Principle 1**, Bruno's confidentiality level becomes higher than the confidentiality level of Carl. In the above example, we were able to understand which read access has a more important effect on the subject confidentiality level by simply comparing the levels of objects read. However, as we show below, such a technique is no longer sufficient when objects confidentiality levels are the same.

Example 2. Let us extend Example 1 by considering an additional subject Sabrina and an additional object o_2. The confidentiality levels are given in Table 2 as follows: $sl(Sabrina) = 1$ and $ol(o_2) = 4$.

Suppose that Bruno has read objects o_1 and o_2 while Sabrina has only read o_2. Now, if we were to determine which of these two subjects has a higher confidentiality level, then according to **Principle 2**, we are likely to conclude that Bruno's confidentiality level has become higher than Sabrina's confidentiality level. This is because the number of objects that Bruno has read with a confidentiality level 4 (2 objects) is higher than the number of objects that has been read by Sabrina and having the same confidentiality level (1 object).

Example 3. Let us extend Example 2 by considering an additional object o_3 and an additional subject Nadia. The confidentiality levels are given in Figure 2 as follows: $sl(Nadia) = 2$ and $ol(o_3) = 3$.

Suppose a case where Nadia has read o_1 and o_3 while Carl has read o_2 and o_4. Now, if we were to determine which of these two subjects has a higher confidentiality level, then according to **Principle 1** and **Principle 2** we are likely to conclude that Nadia's confidentiality level has become higher than Carl's confidentiality level.

This is because the two subjects have read o_1 and o_2 which have the same confidentiality levels but Nadia has also read object o_3 having a confidentiality level 3. The latter is higher than the confidentiality level 2 of o_4 which has been read by Carl.

Remark 1 (from Examples 1, 2 and 3)

On the basis of our definitions, assumptions and principles, we propose the following confidentiality level assessment method:

- Always apply **Principle 1**.
- Whenever higher confidentiality levels of objects read by the subject are the same, apply **Principle 2**.

The following definitions will be needed to formalize the principles of Remark 1:

Multisets

Multisets are like sets, but allow multiple occurrences of identical elements. Formally, a multiset is a pair (L, m) where L is the support set and $m : L \rightarrow N$ is the multiplicity function. In the multiset (L, m), the level x appears m(x) times. For example, {1,2,1,2, 2,4} is the multiset ({1,2,3,4,5}, m) where m is the function such that m (1) = 2, m (2) = 3, m (3) = 0, m (4) = 1 and m (5) = 0.

Multisets order

We deal with finite multisets whose support set is a set of natural numbers. We suppose the usual ordering relationship of natural numbers. We define two relations: $>_{mul}$ and $=_{mul}$ on multisets in the following way: we write the multisets as ordered sequences and then we compare them according to the usual lexicographical order [7].

Example. To compare the multisets {4,4,5,1} and {4,3,2,3,1,5}, we compare the ordered sequences (5,4,4,1) and (5,4,3,3,2,1). Since (5,4,4,1) is lexicographically greater than (5,4,3,3,2,1), it follows that {4,4,5,1} $>_{mul}$ {4,3,2,3,1,5}. In the same way, we can see that {3,3,4,0} $=_{mul}$ {3,4,0,3}.

It can be seen that we can compare subjects confidentiality levels, in terms of their previous read accesses by adhering to the principles of Remark 1. Essentially, these principles can be formalized as follows:

Table 3. Formal definition of Remark 1

1.	$sl(s) > sl(s')$ if $CKSL^+(s) >_{mul} CKSL^+(s')$
2.	$sl(s) = sl(s')$ if $CKSL^+(s) =_{mul} CKSL^+(s')$

Based on Remark1, we obtain the following ordering of subjects confidentiality levels in Examples 1, 2 and 3: sl(Bruno) > sl(Nadia) > sl(Carl) > sl(Sabrina). This is because $CKSL^+$(Bruno) = {4,4} $>_{mul}$ $CKSL^+$(Nadia) = {4,3} $>_{mul}$ $CKSL^+$(Carl) = {4,2} $>_{mul}$ $CKSL^+$(Sabrina) = {4}.

3.1.1.Consideration of information association and information aggregation for subject levels assessment

In the previous section we considered information that can flow from objects to subjects. In this section, we consider information that could be inferred from previous read accesses. This concept is known in the literature as the inference problem [2]. An inference presents a security breach if more highly classified information can be

inferred from less classified information [2]. There are two important cases of the inference problem:

- Information aggregation problem, in the context of access control systems, occurs whenever there is a collection of data items that can be known by a subject or can be stored by an object and that is classified at a higher level than the levels of individual data items by themselves.
  Example. The content of a medical file is Secret, but the aggregate information concerning all the medical files is Top Secret. The blueprint of a single piece is Secret, but the blueprint of a whole artifact is Top Secret.
- Information association problem, in the context of access control systems, occurs whenever two or more values that can be known by a subject or can be stored by an object are classified at a higher level than the classification of either value individually.
  Example. The list consisting of the names of all employees is unclassified and the list containing all employees social insurance numbers are secret, while a combined list giving employee names with their social insurance numbers is Top secret.

We define the following functions that their values are determined by system administrators as a result of enterprise policies where PL is the power multiset of L, PO is the power set of O and N is the set of natural numbers:

- Agg : $N \times L \rightarrow L$ formally represents the assignment of confidentiality levels to information that could be inferred after a number of accesses to a set of objects having the same confidentiality level.
  Example. Agg(3,2) = 4 means that the confidentiality level of the information that could be inferred from 3 accesses to level 2 is 4.
- Ass : $PO \rightarrow L$ formally represents the assignment of confidentiality levels to information that could be obtained from a set of objects.
  Example. Ass({o, o', o''}) = 3 means that the confidentiality level of the information that could be inferred from objects o, o' and o'' is 3.

In order to apply our approach for subject confidentiality level assessment presented in the previous section and to consider at the same time, cases where more highly classified information could be inferred from less classified information, we define the following functions:

- Agg_l : $PL \rightarrow PL$ formally represents the multiset of confidentiality levels that could be inferred as a result of information aggregation.
  Example. Agg_l(CKSL(s)) = {4,4} means that two items of information of confidentiality levels 4 could be inferred from the multiset of levels CKSL(s).
- Ass_l : $PO \rightarrow PL$ formally represents the multiset of confidentiality levels of information inferred as a result of information association.
  Example. Ass_l(CKS(s)) = {4,2} means that CKS(s) contains objects such that, when associated, information of confidentiality levels 4 and 2 could be inferred.
- For $s \in S$, CKSLA(s) is the multiset containing the multiset CKSL(s) and levels of information inferred from CKSL(s) and CKS(s). More formally, $CKSLA(s) = CKSL(s) \cup Agg\_l(CKSL(s)) \cup Ass\_l(CKS(s))$.
- $CKSLA^{+}(s)$ is the submultiset of CKSLA(s) having values equal to or greater than sl(s). More formally, $CKSLA^{+}(s) = \{l \in CKSLA(s) \mid l \geq sl(s)\}$.

Example. Let us assume that $Ass(o_4,o_7) = 3$, $Agg(2,1) = 4$, $CKS(Claude) = \{o_6,o_4,o_7\}$, $CKSL(Claude) = \{2,1,4,1\}$ and $sl(Claude) = 2$.
Thus, we have the following:
$Agg\_l(CKSL(Claude)) = \{4\}$, $Ass\_l(CKS(Claude)) = \{3\}$.
$CKSLA(Claude) = \{2,1,4,1,4,3\}$, $CKSLA^+(Claude) = \{2,4,4,3\}$.

Remark 2

We can compare the confidentiality levels of subject s and subject s', in terms of their previous read accesses and information inferred from these accesses, by comparing $CKSLA^+(s)$ with $CKSLA^+(s')$. This remark can be formalized as follows:

Table 4. Formal definition of Remark 2

1.	$sl(s) > sl(s') \text{ if } CKSLA^+(s) >_{mul} CKSLA^+(s')$
2.	$sl(s) = sl(s') \text{ if } CKSLA^+(s) =_{mul} CKSLA^+(s')$

3.1.2.Subject level assessment when a write access is requested

In this section, we present a subject level assessment approach when a write access is requested. We consider the information that could be inferred from information that can be known by a subject and information that can be stored in the requested object. Whenever a subject requests a write access to an object, the history of its accesses is analyzed to determine whether the information that can be stored in the requested object, correlated with information that can be known by the subject, could result in an inference generating higher level information [1]. If the possibility of an inference arises, the subject's confidentiality level used to determine the access decision for the request in question should be recalculated by considering this possibility. We then need to constitute a new set of objects confidentiality levels where the new levels added are higher than the subject's confidentiality level. In order to apply our approach, we define the following functions:

- $sol : S \times O \rightarrow L$ formally represents the assignment of a confidentiality level to a subject when it requests to write in a particular object.
- For $s \in S$ and $o \in O$, CKSSL(s, o) is the multiset of levels of objects o' for which CS(o, o') is true and levels of objects o'' for which CK(s, o'') is true, $o' \neq o''$, in addition to the level assigned by the administrator to the subject s. More formally, $CKSSL(s, o) = \{ol(o') \mid CS(o, o') = true\} \cup \{ol(o'') \mid CK(s, o'') = true\} \cup sl(s) \setminus \{ol(o') \mid CS(o, o') = CK(s, o')\}$. The symbol \ denotes set difference.
- For $s \in S$ and $o \in O$, CKSS(s, o) is the set of objects o' for which CK(s, o') is true and the set of objects o'' for which CS(o, o'') is true , $o' \neq o''$. More formally, $CKSS(s, o) = \{o' \mid CS(o, o') = true\} \cup \{o'' \mid CK(s, o'') = true\} \setminus \{o' \mid CS(o, o') = CK(s, o')\}$.

We defined the functions CKSSL(s, o) and CKSS(s, o) to avoid the consideration of cases where the same object can be known by subject s and can be stored by object o. This is to be in accordance with our definitions in section 3 when we consider multiple accesses by the same subject to the same object only once.

- For s ∈ S and o ∈ O, CKSSLA(s, o) is the multiset of levels in CKSL(s) and the levels of information inferred from CKSSL(s, o) and CKSS(s, o). More formally, CKSSLA(s, o) = CKSL(s) ∪ Agg_l(CKSSL(s, o)) ∪ Ass_l(CKSS(s, o)).
- $CKSSLA^{+}(s, o)$ is the submultiset of CKSSLA(s, o) having values equal to or greater than sl(s). More formally, $CKSSLA^{+}(s) = \{l \in CKSSLA(s) \mid l \geq sl(s)\}$.

Example. Suppose that Nadia requests to write in o_3.
Let us assume that sl(Nadia) = 2, CKSL(Nadia) = {2,1,1,1}, CKS(Nadia) = $\{o_7,o_6,o_5\}$, CSSL(o_3) = {3,1,1,4}, CSS(o_3) = $\{o_3,o_7,o_8,o_1\}$, Agg(4,1) = 4 and Ass(o_8,o_6) = 3.
Thus, we have the following:
CKSSL(Nadia, o_3) = {3,1,4,2,1,1,1},CKSS(Nadia, o_3) = $\{o_3,o_8,o_1,o_7,o_6,o_5\}$.
Agg_l(CKSSL(Nadia, o_3)) = 4, Ass_l(CKSS(Nadia, o_3)) = 3.
CKSSLA(Nadia, o_3) = {2,1,1,1,3,4}, $CKSSLA^{+}$(Nadia, o_3) = {2,3,4}.

Remark 3

We can compare confidentiality levels of subject s and subject s' when they request write access to an object o, in terms of their previous read accesses, information that can be known or inferred from previous accesses or the current one, by comparing $CKSSLA^{+}(s, o)$ with $CKSSLA^{+}(s', o)$. This remark can be formalized as follows:

Table 5. Formal definition of remark 3

1.	$sol(s,o) > sol(s',o)$ if $CKSSLA^{+}(s,o) >_{mul} CKSSLA^{+}(s',o)$
2.	$sol(s,o) = sol(s',o)$ if $CKSSLA^{+}(s,o) =_{mul} CKSSLA^{+}(s',o)$

3.2. Access history and information flow-based object confidentiality level assessment

The confidentiality levels of objects may be assessed by referring to information that can be stored in the objects. In this section, we say that object o can be stored in object o' instead of saying that information from object o can be stored in object o'. Our method is designed to satisfy the following requirements:

Principle 4: The confidentiality level of an object increases as the object can store objects having confidentiality levels equal to or higher than its own, i.e. the confidentiality level of objects that it can store increases.

Principle 5: The confidentiality level of an object increases as the object can store greater *number* of objects having confidentiality levels equal to or higher than its own, even if their confidentiality level does not increase.

Principle 6: For objects that have not yet been written by subjects with higher confidentiality levels, the confidentiality level is set to a default value. This can be determined by the system administrator.

Running example. We describe a scenario for motivating our history based objects level assessment approach. The objective of our work in this section is to order object confidentiality levels. Hence, if CS(o, o') is true, we only cite examples where ol(o') ≥ ol(o) because objects confidentiality levels change only when objects can store information from objects having equal to or higher confidentiality level than their own.

We now give examples that motivate our technique for object confidentiality level assessment that is primarily based on the confidentiality levels of objects that can be stored in objects.

Example 4. Suppose a first case where object o_1 can be stored in object o_5 and a second case where object o_4 can be stored in object o_6. We have the following from Table 2: $ol(o_1) = 4$, $ol(o_4) = 2$, $ol(o_5) = 1$ and $ol(o_6) = 1$.

According to **Principle 4**, the fact that object o_1 can be stored in object o_5 makes o_5's confidentiality level higher than the confidentiality level of object o_6 where object o_4 can be stored. This is simply because the confidentiality level of object o_1 is higher than the confidentiality level of o_4. In the above example, we were able to understand which object has a more important effect on the object's confidentiality level by simply comparing the levels of those two objects that can be stored in each object. However, as we show below, such a technique is no longer sufficient when the confidentiality levels of objects are the same.

Example 5. Let us extend Example 4 by considering the following objects o_2 and o_7. The confidentiality levels are given in Table 2 as follows: $ol(o_7) = 1$ and $ol(o_2) = 4$. Suppose that objects o_1 and o_2 can be stored in o_5 and o_2 can be stored in o_7.

Now, if we were to determine which of these two objects (o_5 and o_7) has a higher confidentiality level, then according to **Principle 5** we are likely to conclude that o_5's confidentiality level is higher than o_7's confidentiality level. This is because the number of objects that can be stored in o_5 with a confidentiality level of 4 (2 objects) is higher than the number of objects with the same confidentiality level that can be stored in o_7.

Example 6. Let us extend Example 5 by considering additional objects o_3 and o_8. The confidentiality levels are given in Table 2 as follows: $ol(o_3) = 3$ and $ol(o_8) = 1$.

Suppose a case where objects o_1 and o_3 can be stored in o_8 and, at the same time, o_4 can be stored in o_6. Now, if we were to determine which of these two objects has a higher confidentiality level, then according to **Principles 4** and **5** we are likely to conclude that o_8's confidentiality level is higher than o_6's confidentiality level. This is because o_8 can store object o_3 having a confidentiality level 3 which is higher than the confidentiality level of o_4 that can be stored in o_6.

Remark 4 (from Examples 4, 5 and 6)

The previous principles and examples suggest the following method for assessing confidentiality levels of objects:

- Always apply **Principle 4**.
- Whenever higher confidentiality levels of objects which can be stored in the object, are the same, apply **Principle 5**.

It can be seen that we can compare objects confidentiality levels, in terms of the objects that can be stored in them by adhering to the principles of Remark 4. Essentially, these principles can be formalized as follows:

Table 6. Formal definition of Remark 4

1. $ol(o) > ol(o')$ if $CSSL^+(o) >_{mul} CSSL^+(o')$
2. $ol(o) = ol(o')$ if $CSSL^+(o) =_{mul} CSSL^+(o')$

Based on Remark4, we obtain the ordering of objects confidentiality levels in Examples 4, 5 and 6 as follows: $ol(o_5) > ol(o_8) > ol(o_6) > ol(o_7)$. This is because $CSSL^+(o_5) = \{4,4\} >_{mul} CSSL^+(o_8) = \{4,3\} >_{mul} CSSL^+(o_7) = \{4\} >_{mul} CSSL^+(o_6) = \{2\}$.

3.2.1. Consideration of Information association and Information aggregation for object level assessment

In this section, we consider the inference of information in addition to information obtained from previous accesses. In order to apply our approach for objects confidentiality levels assessment and to consider cases where more highly classified information could be inferred from less classified information, we define the following functions:

- For o ∈ O, CSSLA(o) is the multiset containing the multiset CSSL(o) and levels of information inferred from CSSL(o) and CSS(o). More formally, CSSLA(o) = CSSL(o) ∪ Agg_l(CSSL(o)) ∪ Ass_l(CSS(o)).
- $CSSLA^{+}(o)$ is the submultiset of CSSLA(o) having values equal or greater than ol(o). More formally, $CSSLA^{+}(o) = \{l \in CSSLA(o) \mid l \geq ol(o)\}$.

Example. Let us assume that $CSS(o_3) = \{o_3,o_6,o_4,o_7\}$, $CSSL(o_3) = \{3,1,2,1\}$, Ass $(o_4, o_7) = 3$, Agg $(2, 1) = 4$ and $ol(o_3) = 3$.
Thus, we have the following:
$Agg\_l(CSSL(o_3)) = \{4\}$, $Ass\_l(CSS(o_3)) = \{3\}$.
$CSSLA(o_3) = \{3,1,2,1,4,3\}$, $CSSLA^{+}(o_3) = \{3,4,3\}$.

Remark 5

We can compare confidentiality levels of object o and object o' in terms of objects which can be stored in them and information inferred as a result of previous accesses, by comparing $CSSLA^{+}(o)$ with $CSSLA^{+}(o')$. This remark can be formalized as follows:

Table 7. Formal definition of remark 5

1.	$ol(o) > ol(o')$ if $CSSLA^{+}(o) >_{mul} CSSLA^{+}(o')$
2.	$ol(o) = ol(o')$ if $CSSLA^{+}(o) =_{mul} CSSLA^{+}(o')$

3.2.2. Object level assessment when a read access is requested

In the previous section, we considered information that could be inferred from information which can be stored in an object. In this section, we consider information that could be inferred from information which can be known by a subject and information which can be stored in the requested object. Whenever a subject requests a read access to an object, the history of its accesses is analyzed to determine whether the information that can be stored in the requested object, correlated with information that can be known by the subject, could result in an inference generating high level information although the information written had a low level. Therefore if an inference arises, the object's confidentiality level to be used to determine the access decision for the request in question, should be recalculated by considering this inferred information. In order to apply our approach, we define the following functions:

- osl : O × S → L formally represents the assignment of a confidentiality level to an object when a subject requests to read it.
- For o ∈ O and s ∈ S, CSSSLA(o, s) is the multiset of levels in CSSL(o) and levels of information inferred from CKSSL(s, o) and CKSS(s, o). Formally, CSSSLA(o, s) = CSSL(o) ∪ Agg_l(CKSSL(s, o)) ∪ Ass_l(CKSS(s, o)).

- CSSSLA$^+$(o, s) is the submultiset of CSSSLA(o, s) having values equal to or greater than ol(o). More formally, CSSSLA$^+$(o, s) = {l ∈ CSSSLA(o, s) | l ≥ ol(o)}.

Example. Suppose that Nadia requests to read o_3 and let us assume that ol(o_3) = 3, CKSL(Nadia) = {2,1,1,1}, CKS(Nadia) = {o_7,o_6,o_5}, CSSL(o_3) = {3,1,1,4}, CSS(o_3) = {o_3,o_7,o_8,o_1}, Agg(4,1) = 4 and Ass(o_8,o_6) = 3.
Thus, we have the following:
CKSSL(Nadia, o_3) = {3,1,4,2,1,1,1}, CKSS(Nadia, o_3) = {o_3,o_8,o_1,o_7,o_6,o_5}.
Agg_l(CKSSL(Nadia, o_3)) = 4, Ass_l(CKSS(Nadia, o_3)) = 3.
CSSSLA(o_3, Nadia) = {3,1,1,4,3}, CSSSLA$^+$(o_3, Nadia) = {3,4,4}.

Remark 6
We can compare confidentiality levels of object o and object o' when a subject s requests read access to them, in terms of objects stored in them, information inferred from these objects and information that could be inferred from information known by the subject and information stored in the object, by comparing CSSSLA$^+$(o, s) with CSSSLA$^+$(o', s). This remark can be formalized as follows:

Table 8. Formal definition of Remark 6

1.	osl(o,s)> osl(o',s)if CSSSLA$^+$(o,s)>$_{mul}$ CSSSLA$^+$(o',s)
2.	osl(o,s)= osl(o',s)if CSSSLA$^+$(o,s)=$_{mul}$ CSSSLA$^+$(o',s)

4. Access history and information flow-based integrity levels assessment

In the previous sections, we have presented an approach for subjects and objects confidentiality levels assessment. This approach is based on the idea that confidentiality levels of subjects and objects increase when information can flow down to them. In this section we will present a set of principles to assess the integrity levels. Our approach is based on the idea that integrity levels of subjects and objects decrease when information can flow up to them [6]. In other words subjects can decrease their integrity levels as they can know information from lower levels and objects can decrease their integrity levels as they can store information from lower levels. The number of accesses is another factor to be considered when assessing subject and object integrity levels.

Our history and information flow-based subject's integrity level assessment approach is based on the principles that are contextually defined below:

- Always apply **Principle 7** that is, the integrity level of a subject decreases as the subject reads (can know) objects having integrity levels equal to or lower than its own.
- Whenever lower integrity levels of objects read by the subject are the same, apply **Principle 8** that is, the integrity level of a subject decreases as the object can store greater *number* of objects having integrity levels equal to or lower than its own.
- For subjects that has no history of reading objects, apply **Principle 9** that is, the integrity level is set to a maximum / default value. This can be determined by the system administrator.

Our history and information flow-based object's integrity level assessment approach is based on the principles defined below:

- Always apply **Principle 10** that is, the integrity level of an object decreases as the object can store objects having integrity levels equal to or lower than its own.
- Whenever lower integrity levels of objects which can be stored in the object are the same, apply **Principle 11** that is, the integrity level of an object decreases as the object can store greater *number* of objects having integrity levels equal to or lower than its own.
- For objects that have not yet been written by subjects with lower integrity levels, apply **Principle 12** that is the integrity level is set to a maximum / default value. This can be determined by the system administrator.

5. Discussion and Related work

To the best of our knowledge, the assessment of subjects and objects security levels, by considering information flow, has been presented via two models which were introduced in [3]: the High-water mark which predates the Bell Lapadula model [4] and the Low-water mark which is an extension of the Biba model [5].

Under the High-water mark, when a subject reads an object of higher confidentiality level, the object's confidentiality level is assigned to the subject and when a subject writes in an object of lower confidentiality level, the subject's confidentiality level is assigned to the object. These two properties are similar to **Principle 1** and **Principle 4** presented in this paper. However, in our approach, the highest confidentiality level will not automatically be assigned to subjects and objects. This is because we consider all higher confidentiality levels of objects which can be stored in the object and all higher confidentiality levels of objects which have been read by the subject in addition to the number of accesses.

Under the Low-water mark, when a subject reads an object of lower integrity level, the object's integrity level is assigned to the subject and when a subject writes in an object of higher integrity level, the subject's integrity level is assigned to the object. These two properties are similar to **Principle 7** and **Principle 10** presented in this paper. However, in our approach, the lowest integrity level will not automatically be assigned to subjects and objects. This is because we consider all lower integrity levels of objects which can be stored in the objects and all lower integrity levels of objects which have been read by the subject in addition to the number of accesses. In both cases, our approach is far more sophisticated than these previously known approaches.

6. Conclusion

The main contribution of this paper is a framework that includes a history and information flow-based approach for subjects and objects level assessment. This approach is based on past accesses and considers information which can flow from previous accesses, as well as information that can be inferred from previous accesses (information aggregation and information association). Information that could be inferred from information that can be known by a subject and information that can be

stored in an object are also considered. Towards this end, we have presented several examples that justify our approach in intuitive terms. We have also presented a formal definition of our approach.

To the best of our knowledge, our work represents one of the few attempts in the literature to conduct a history and information flow-based approach for entity security levels assessment. We have shown that our approach is a substantial improvement with respect to the previously known approaches of the high and low watermark models. This approach is valuable in Web and Cloud environments where there will be many continuously evolving information flows, since our methods can be invoked dynamically as the information moves between subjects and objects.

As mentioned in Section 1, our ultimate goal is to develop a framework for estimating security levels. This requires us to extend the work reported in this paper by defining mathematical formulas which capture the presented principles.

Acknowledgment. This research was funded in part by grants of the Natural Sciences and Engineering Research Council of Canada.

References

1. Sandhu, R.S and Jajodia, S, Data and database security and controls. Security and controls handbook of Information Security Management, Auerbach Publishers (1993).
2. Sandhu, R.S, Lattice-based access control models. Computer 26(11) (1993).
3. Weissmann, C, Security controls in the ADEPT-50 timesharing system. AFIPS Conference Proceedings FJCC (1969).
4. Bell, D.E and LaPadula, L.J, Secure Computer Systems: Mathematical Foundations. MITRE Corporation (1973).
5. Biba, K, Integrity considerations for secure computer systems. Technical Report TR-3153, MITRE Corporation (1977).
6. Logrippo, L. Logical Method for Reasoning about Access Control and Data Flow Control Models. To appear in the Proc. of the 7th International Symposium on Foundations and Practice of Security (2014).
7. Dershowitz, N and Manna, Z, Proving termination with multiset orderings, Communications of the ACM 22 (8), (1979).