International Journal of Safety and Security Engineering
Vol. 11, No. 5, October, 2021, pp. 529-535
Journal homepage: http://iieta.org/journals/ijsse

Coupling of Inference and Access Controls to Ensure Privacy Protection

Jihane El Mokhtari[1,2*], Anas Abou El Kalam[3], Siham Benhaddou[2], Jean-Philippe Leroy[1]

[1] LISER Laboratory, IPI, Paris 75010, France
[2] LRI Laboratory, ENSEM, Casablanca 20000, Morocco
[3] TIM Laboratory, ENSAM, Marrakesh 40000, Morocco

Corresponding Author Email: jihane.elmokhtari-etu@etu.univh2c.ma

https://doi.org/10.18280/ijsse.110504

Received: 7 August 2021
Accepted: 2 October 2021

Keywords:
access control, inference control, multidimensional analysis, privacy

ABSTRACT

This article is devoted to the topic of coupling access and inference controls into security policies. The coupling of these two mechanisms is necessary to strengthen the protection of the privacy of complex systems users. Although the PrivOrBAC access control model covers several privacy protection requirements, the risk of inferring sensitive data may exist. Indeed, the accumulation of several pieces of data to which access is authorized can create an inference. This work proposes an inference control mechanism implemented through multidimensional analysis. This analysis will take into account several elements such as the history of access to the data that may create an inference, as well as their influence on the inference. The idea is that this mechanism delivers metrics that reflect the level of risk. These measures will be considered in the access control rules and will participate in the refusal or authorization decision with or without obligation. This is how the coupling of access and inference controls will be applied. The implementation of this coupling will be done via the multidimensional OLAP databases which will be requested by the Policy Information Point, the gateway brick of XACML to the various external data sources, which will route the inference measurements to the decision-making point.

1. INTRODUCTION

Respecting the confidentiality and privacy of complex systems' users is arguably one of the main requirements that every system must satisfy. Access Control and Inference Control are two mechanisms used to ensure this fundamental requirement and protect sensitive data from unauthorized disclosure. Although these two mechanisms share the same goal, access control is more widely used in research work due to its efficiency and accuracy in calculations, as well as its accessibility and adaptability to different environments. Nevertheless, it remains insufficient to foresee all the information leaks that may occur, via discrete channels, in full compliance with all static access control rules. Moreover, the management of dynamic inferences requires expensive inference control, which has prompted multiple attempts to substitute inference control with access control; in this case, the definition of access rights must be done very carefully so that sensitive data is effectively protected against indirect access.

Despite efforts in this direction, all attempts to extend access control to incorporate inference control have failed to provide an effective approach, especially in complex systems. The coupling of access and inference mechanisms is therefore necessary. Several works aiming at a joint implementation of access and inference controls already exist; they will be studied and discussed in section 3 of this paper. We will devote our efforts in this work to defining an inference control mechanism to be incorporated into our access control policy. In previous works, the choice of the PrivOrBAC access control model was made to meet all identified privacy protection requirements. The presentation of PrivOrBAC is done via the PrivUML metamodel proposed by El Mokhtari et al. [1]. For the implementation of the PrivUML model, the XACML architecture has been adopted and the XACML language has been evolved to allow the transformation of PrivUML [2]. Other aspects of privacy protection have been covered thanks to the XACML standard as explained in the study [2]. However, despite all the measures taken, the risk of data disclosure remains present by associating several accessible data to deduce non-accessible information. In this article, we will explain how inference control will be provided through multidimensional analysis to cover this risk, and how this inference control will be coupled with access control and participate in access decision making.

The remainder of this article is organized into four sections. Section 2 provides an overview of our previous work. The review of related work is presented in section 3. Section 4 is dedicated to the presentation of our proposal for the application of inference control in our security policy, and finally section 5 is devoted to the case study which applies our approach.

2. PREVIOUS WORKS

Our work is focused on protecting Privacy in complex systems. With the view of putting in place a mechanism capable of guaranteeing user privacy, we have respected the recommendations of the Model Driven Architecture (MDA) approach. At the Common Independent Model (CIM) level, we have adopted the PrivOrBAC [3] access control model,

which is an extension of the OrBAC [4] model adjusted to cover all privacy protection requirements formalized by Abou El Kalam et al. [5]. The transition to the second level of MDA is done by translating the CIM model into a Platform Independent Model (PIM), and requires finding the modeling tool capable of integrating all our Privacy requirements into the target model. We presented our PrivUML metamodel, which is an extension of UML enriched by the notions of access modalities, object view hierarchy, purpose and consent [1].

Following the MDA approach, PrivUML must undergo a transformation into a Platform Specific Model (PSM), and we ultimately chose the OASIS XACML standard [6]. We explained in the study [2] how we mapped the conceptually-similar model elements between PrivUML and the XACML language that we have evolved to materialize the set of privacy protection requirements presented by PrivUML. We also showed how we took advantage of the architecture and the XACML language to dynamically integrate other aspects of privacy protection (anonymization, pseudonymisation, etc.).

So far, we have presented how we came to define a XACML policy to strengthen privacy protection and ensure all privacy requirements in a system in accordance with the MDA approach. The management of this policy is static at this stage, and it is in the study [7] that we presented how we can automate the management of our policy by relying on smart contracts and the WS-Agreement specification [8].

3. RELATED WORKS

Several works have approached the subject of inference in access control in very different ways. Some have exploited the possibility of substituting inference control with access control [9-11]. Another way to deal with inference in access control is to apply them together. Several studies have explored this logic and their starting point is the observation that sensitive data can be revealed indirectly despite the application of access control rules. This observation gave rise to several lines of interpretation. The introduction of semantics into access control mechanisms is one of the axes studied. Paci and Zannone [12] claim that ignoring the semantic relationship between data when specifying access control policies is at the origin of the inference. They conducted a comparative study of the different evaluation functions used by access control models, depending on their data structures, to determine the strategies to apply to an access request. They then proposed an access control model based on a semantic approach that exploits knowledge about the application domain by structuring it in a hierarchical data model. The protection of sensitive data from inference is guaranteed by defining the rules for authorizations and access denials based on the semantic relationships between the protected data and its ancestors / predecessors that may lead to its disclosure. Despite this, the inference is still possible with this proposition due to the vertical evaluation of the propagation; the accumulation of two or more data that are individually harmless can lead to an inference of sensitive data.

Auxilia and Raja [13] have chosen to rely on semantics in the design of the Knowledge-Based Security Model (KBSM) that allows joint control of access and inference. The four components of the KBSM model are an ontology base (for subject, resources, and actions), a policy base, an inference engine, and a policy engine. In KBSM, the inference engine intervenes first by submitting the data relating to the access request (subject, resource and action), collected by the user interface, to an Inference Control against the corresponding ontologies. The inference engine then calls on the policy engine to process the access request according to the rules established in the policy base. The designers of KBSM implemented this inference control coupled with access control for a different purpose than ours. They want to reduce the number of rules to be saved in the rule base; a subject *s* having the right to exercise an action *a* on an object *o1* according to the rule *R1(s, o1, a)*, can also exercise the same action *a* on another object *o2* in accordance with the existing inference between *o1* and *o2* without a rule *R2(s, o2, a)* being explicit. KBSM is therefore unable to meet our needs and prevent the inference of one sensitive piece of data by combining several others.

Another approach that seeks to ensure access control while preventing inference attacks by relying on semantics is proposed by Jebali et al. [14]. This approach, intended for application in Cloud Computing, proposes dividing sensitive data into a set of partitions to be stored separately in the servers of cloud service providers. Vertical data partitioning takes into account the semantic relationships between attributes and user roles, and aims to maximize intra-dependency within a single partition while minimizing the interdependence between attributes in separate partitions. The Inference Control in this approach relies on identifying functional dependencies between attributes and generating join strings that are cut at a single point representing a confidentiality constraint. Although this approach is interesting on account of its application of control over join points allowing the inference channels to be broken, it does not consider all possible semantic dependencies as sources of inference; Only functional and probabilistic dependencies are taken into account, while include, join, and multi-valued dependencies are not.

In the same context of using semantics in the prevention of inference on data, the authors of the studies [15-17] presented a succession of works in which they first built a directed hypergraph schematizing the dependencies between the data. The set of operations that subjects can perform on objects according to the Access Control List (ACL) is expressed as colored vertices, thus constructing a coloration list on the constructed hypergraph. They then used this coloration list to identify hidden channels of inference with read / write operations. Finally, they proposed historicizing the operations performed by the subjects using the blockchain as a logging system. The weak point of this proposal lies in the difficulty of optimizing the coloration of hypergraphs; this difficulty increases with the complexity of the system and the volume of data it handles. Hypergraphs' coloring continues to be the subject of research which attempts to provide algorithms capable of optimizing the coloring of large volumes of data in the presence of constraints and conflicts on that data. In addition, the use of blockchain as a logging system is not suitable for decision making. Blockchain is certainly an excellent technology for distributed and secure data storage, but the high write and read time prevents its use in decision-making systems, in this case the decision-making of access.

4. INFERENCE CONTROL INTEGRATION IN THE PRIVORBAC MODEL

We have taken care in our previous work to cover several

key aspects of privacy protection. In the present article, we focus on Inference Control. This section is dedicated to the presentation of our proposal for coupling access and inference controls into the access decision-making mechanism. Our objective is to prevent a subject from accumulating several data to which he/she has access in order to prevent him/her from combining them to deduce information to which he/she does not have direct access. The starting point of our proposal is the mastery of the semantic relations linking the data. The role of design analysts is to identify the different combinations of data which can lead to inference based on their knowledge of the application domain. These combinations relate to the domain and are therefore generic; they constitute the system's set of inference channels. User personalization will occur afterwards according to the list of data that he/she decides to classify as private. When a subject S formulates a request for access to data D belonging to the owner O, the access control rules are consulted first to ensure that D can be accessed by S outside of inference. The list of O's own inference channels is then consulted to verify whether D is part of one or more channels. If D is recognized as a datum capable of triggering an inference, it is necessary to check whether the accumulation of D with all the data previously accessed by S and belonging to the same channels as D is sufficient to lead to an inference. This level of control therefore requires logging of the accesses of each subject. We are therefore obliged to manage a large amount of data for decision-making purposes, taking into account several criteria simultaneously. In this article, we propose using multidimensional analysis to set up inference control. We shall detail the different steps of our proposal in this section.

4.1 Construction of the list of inference channels by user (LICU)

The list of functional dependencies determined in the design phase of a system constitutes our repository for building the list of inference channels for each user. The projection of the data that a user wishes to keep private on the semantic repository allows us to build the matrix of inference channels relating to this user:

Sensitive Data	*L*	*I*	*C*	*U*
*SD*1	*A*	*B*	*C*	*H*
*SD*1	*A*	*B*	*E*	*L*
*SD*2	*A*	*C*	*F*	*K*
*SD*3	*C*	*L*	*E*	*O*
*SD*4	*R*	*O*	*F*	*P*

In this example, user *O1* has chosen to protect data *SD1, SD2, SD3 and SD4*. The above matrix models the different inference channels leading to each of these data. When a request for access to data *B* is made by a subject *S1*, and following the authorization granted by the non-inference access control, the LICU matrix is consulted. Two inference channels are therefore identified:

Sensitive Data	*L*	*I*	*C*	*U*
*SD*1	*A*	*B*	*C*	*H*
*SD*1	*A*	*B*	*E*	*L*
*SD*2	*A*	*C*	*F*	*K*
*SD*3	*C*	*L*	*E*	*O*
*SD*4	*R*	*O*	*F*	*P*

4.2 Inference control by analyzing access history and data weights

At this stage, we need to check if one or more data belonging to the two identified inference channels have been consulted in the past and, if so, ensure that the accumulation of *B* to these data is not sufficient to lead the disclosure of data *SD1* in favour of *S1*. To do this, we need to introduce a new dimension in our matrix which corresponds to the history (HIST) of the data consulted by *S1* as illustrated in Figure 1:

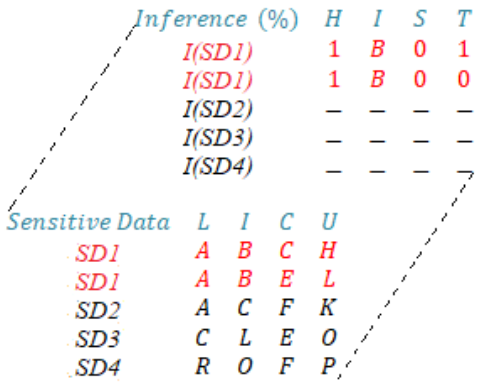

Figure 1. Illustration of the multidimensional matrix of inference channels and access histories

Any data previously consulted by *S1* is marked by **"1"** in the historization dimension. In our example, we see that the access to *B* is not sufficient to traverse one of the two inference channels detected until the deduction of *SD1*. We propose to set up an inference percentage that reflects the portion used for each channel. We can then determine that all of a channel's data have the same weight and that the inference percentage $I(SDi, Sj)$ of the channel bringing the Subject Sj to the sensitive data SDi is calculated by the following formula (1):

$$\mathrm{I(SDi, Sj)} = \frac{100}{\mathrm{n}} * \sum_{\mathrm{k=1}}^{\mathrm{n}} (\mathrm{Dk} = 1) \qquad (1)$$

With *n* corresponding to the total number of data within the channel and *Dk*=1 corresponding to the number of data that have already been consulted. The highest inference percentage to reach *SD1* after consulting *B* is 75%, so access to *B* may be allowed. In reality, one piece of data may be more meaningful than another. If we depart from this observation, and we assume that the weight of *B* is higher than the other data as can be seen in the matrix below, access to *B* will therefore constitute a greater step towards the disclosure of *SD1*:

Inference (%)	*H*	*I*	*S*	*T*
$I(SD1)$	1.25	1.5	0.05	1.2
$I(SD1)$	1.25	1.5	0.1	0.15
$I(SD2)$	–	–	–	–
$I(SD3)$	–	–	–	–
$I(SD4)$	–	–	–	–

We have combined in this matrix the information relating to the access history of each piece of data within the inference channel which might lead a subject Sj to the sensitive data SDi, represented by the integers 0 and 1 before the separator, as well

as the weight assigned to each piece of data, represented by the value following the separator. The sum of the weights of all the data constituting the inference channel must be equal to 1. The new formula (2) for calculating the percentage of inference in the presence of the weights is as follows:

$$I(SDi, Sj) = 100 * \sum_{k=1}^{n} (wk_{Dk=1}) \quad (2)$$

With $wk_{Dk=1}$ corresponding to the weight wk of the data Dk currently or previously consulted. The percentage of inference therefore changes in the presence of weights from 75% to 95%. We can consider setting up a condition which estimates that from a certain percentage, the inference is considered to be triggered, and access to data B is consequently prohibited, even if the inference channel is not fully taken.

4.3 Integration of inference control into security policy

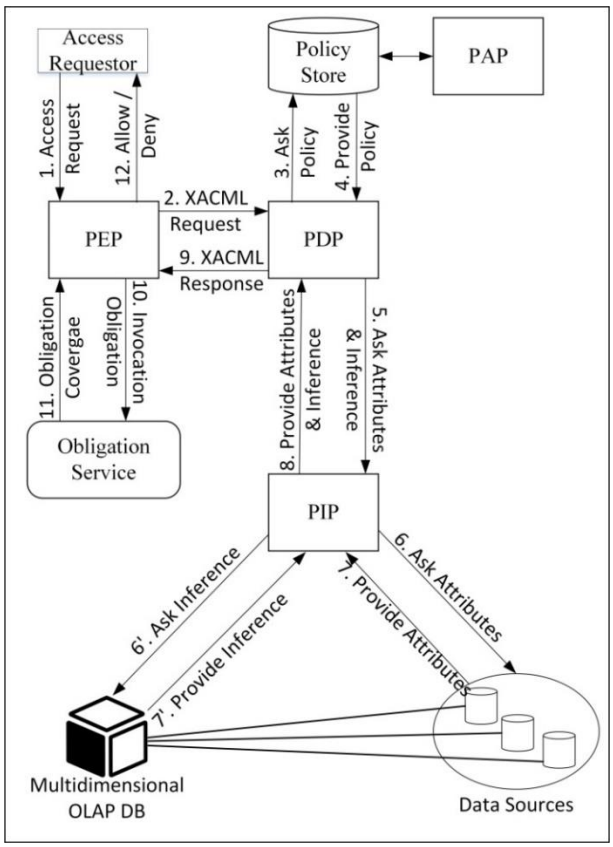

Figure 2. Coupling of access and inference controls in an XACML architecture

We explained in the previous work section how we implemented our security policy based on the XACML architecture. In order to integrate the inference control mechanism into our security policy, we shall put into place the tools necessary to perform this control at the Policy Information Point (PIP) level. PIP is the gateway to external data sources such as LDAP and databases. It provides the Policy Decision Point (PDP) with all the additional information needed to process an access request and make a decision. The PDP, after consulting the access control rules, will call on the PIP to check whether the data to be consulted participates in one or more inference channels and to recover the highest inference percentage corresponding to the most critical channel among those identified. The coupling of access and inference controls consists of including the inference threshold which is not to be exceeded in the case of partial inference in the access rules. We propose setting up two inference thresholds; the first must be linked to the obligation to alert the administrator for access permission while the second will be mentioned as a refusal condition. Our XACML application architecture enriched by the inference mechanism is shown in Figure 2.

As shown in Figure 2, an access request is translated by the Policy Enforcement Point (PEP) into a XACML Request sent to the PDP which, on the one hand, retrieves the security policies administered in the Policy Administration Point (PAP), and on the other hand, requests from the PIP all the attributes necessary for the investigation of the request as well as the value of the inference measured. Traditionally, the PIP retrieves simple attributes corresponding to subjects, resources and environments directly from Data Sources. As for the inference measure, it is provided to the PIP by the multidimensional OLAP database which will consolidate its calculation based on a set of information provided by the data sources. OLAP is the technology chosen for the implementation of the inference mechanism. The following section is intended to explain this technology and present our proposed implementation in more detail. The PIP therefore supplies the PDP with all the expected elements and the latter responds to the PEP with a XACML Response. In the case of rejection or authorization without obligation, the PEP transmits its decision to the access requester. The presence of an obligation in the XACML response, in particular if the first inference threshold is exceeded, requires the PEP to ensure with the Obligation Service that it will be covered before transmitting the authorization to the access requester.

4.4 Implementing inference control

We need to set up a decision information system to perform multidimensional analyses on big and complex data. A multidimensional database and more precisely On-Line Analytical Processing (OLAP) technology is the most appropriate solution to meet our needs. OLAP offers a set of tools for doing online analytical processing requiring complex calculations. The creation of the OLAP multidimensional database is done by importing data from relational databases. OLAP also offers different modeling alternatives through the Relational OLAP (ROLAP), Multidimensional OLAP (MOLAP), Hybrid OLAP (HOLAP) and Hybrid Transactional Analytical Processing (HTAP) products. These systems differ in the way they model data. ROLAP allows for the use of Relational Database Management Systems (RDBMS) through direct access to the data stored in these databases and the construction of multidimensional views. Unlike ROLAP, MOLAP allows the modeling of a multidimensional environment based on OLAP cubes. Each cell in an OLAP cube represents an intersection of dimensions and provides a measure. HOLAP is the combination of the best features of ROLAP and MOLAP and can therefore be used on a multidimensional database as well as a relational database. Finally, HTAP, which made its first appearance in 2014, allows for both analytical and transactional processing. HTAP, housed in the relational database, avoids multiple copies and the need to offload data from operational databases to data warehouses.

Our implementation of inference control can be done by any of the above-mentioned OLAP products. We must first identify the inference channels to which the data to be consulted belongs, then to project the privacy preferences of the data's owner onto the list of identified channels so as to determine those which concern him/her, and to finish measuring the percentage of inference based on the access history of the subject making the request. The three dimensions that we adopt for our analysis are Inference Channels (CI_n), data owners (O_n) and access requesters (S_n). The measure that should result from the intersection of these three dimensions is calculated with formula (2) and shown in Figure 3.

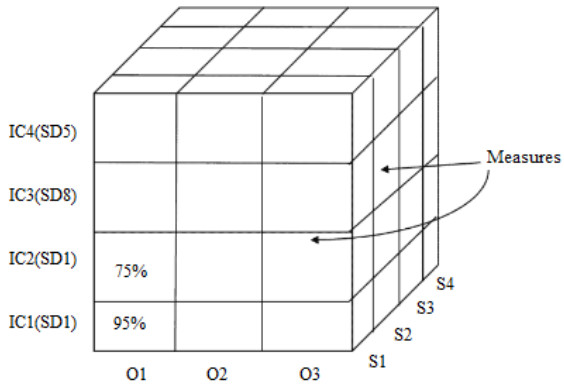

Figure 3. Multidimensional analysis for inference control

The data *B* requested by the subject *S1* participates in the two inference channels leading to the data *SD1* marked as sensitive by the owner *O1*. The subject *S1* which had already consulted other data belonging to the same channels will reach, by consulting *B*, the inference percentages of 95% and 75%; the most critical threshold is retained. If at least one of the inference thresholds specified in the access control rule is lower than this percentage, the administrator will be warned or access to *B* will be refused to *S1* to avoid the deduction of the sensitive data *SD1*.

Let us now suppose that the subject *S1* formulates a new access request relating to another data item *B'* belonging to the owner *O2* who has not privatized any of their data. The inference matrix of *O2* is empty and no inference channel is identified. In this case, the inference check does not return an inference percentage and subject *S1* will be allowed to access *B'*.

5. CASE STUDY

In the healthcare sector, the totality of all members' personal and medical data is managed by the French Health Management System (FHMS). Data owners may opt to designate select information as private. Any access to members' sensitive or non-sensitive information is controlled by the system which grants or denies it according to the predefined access control policy. Despite the careful application of this security policy, private data may be deduced from the combination of several data to which access is completely legal.

Mr. John Doe, who is admitted to the emergency department of H Hospital following a serious accident, is taken in charge by Dr. Schmidt. To retrieve the patient's personal and medical information, Dr. Schmidt formulates a request for access to the FHMS, in which all the information relating to the request is specified. Dr. Schmidt's request is assessed positively in accordance with the access control rules which authorize access to all member information by a doctor other than his/her attending physician in the event of admission to an emergency room following an accident. Bob, meanwhile, is an ER nurse at H Hospital. He makes the same request as Dr. Schmidt, but does not get the same permissions. Although the Context (Emergency) and Reason (Accident) of Bob's request are identical to that of Dr. Schmidt, Bob does not have the "Doctor" role and therefore does not meet all the criteria for access; consequently, he cannot consult patient John Doe's private data.

In our case study, John Doe, who is HIV positive, chose to keep this information private (Sensitive Data). Bob, who does not have access to his private data, cannot consult any data directly indicating the patient's HIV status. He can, however, view all of John Doe's other sensitive unclassified medical data. Bob can therefore see by examining the patient's current treatments that he is being treated with a protein called "Interferon". This protein is used in the treatment of viral diseases (AIDS, Hepatitis, Papilloma Virus, etc.), in Oncology (Sarcoma) or even in preventive treatment. This information alone does not make it possible to deduce with certainty that John Doe is a carrier of HIV, but by combining it with other data (results of some analyses for example), which are also part of the medical data that Bob is able to freely consult, the private information can therefore be revealed. The functional chains leading to information on "Seropositivity" identified upstream by medical practitioners are described in Table 1.

Table 1. Seropositivity functional chains

	P24 Antigen	Interferon	Viral Load	RBCs	T4/T8 Lymphocytes
IC1	1	N/A	N/A	N/A	N/A
IC2	N/A	0.35	0.5	0.05	0.1

These functional chains are potential inference channels depending on the sensitivity of the information of the HIV status to a patient. The first inference channel consists of a single piece of data which is the P24 Antigen; this is a viral analysis revealing Seropositivity, its consultation leads directly to the disclosure of the presence of HIV. Therefore, the weight of this information is equal to 1. The second inference channel consists of four data corresponding to the treatment administered (Interferon), a viral analysis (viral load) and blood tests (RBCs and T4/T8 lymphocytes). Viral Load is the most telling indicator that holds the most weight (0.5). The decrease in red blood cells (weight = 0.05) and lymphocytes (weight = 0.1) can be caused by other infections, so they are not considered to be strong indicators of Seropositivity.

In the following, we present the scenario of access and inference controls coupling following Bob's requests to access John Doe's data:

Step 1: Request for access to "Interferon"

The access control rule authorizes Bob to access this non-sensitive medical data provided that the inference thresholds (75% and 90%) are not exceeded following this consultation. The inference control is therefore triggered and proceeds as

follows:

• Identify the inference channels in which the data to be consulted participates

→ IC2: Interferon, Viral Load, RBCs, T4/T8 Lymphocytes

• Determine if the inference channel (IC2) is active for John Doe

→ The Seropositivity is sensitive for John Doe, therefore IC2 is confirmed active

• Calculate the inference percentage based on the weight of the data (Interferon) and Bob's access history

→ Bob has not yet accessed any other data from channel IC2 and $I(IC2, Bob) = 100 * 0.35 = 35\%$

The calculated inference percentage (35%) has no impact on the access control rule that allows Bob to infer that John Doe is being treated with Interferon.

Step 2: Request for access to "Viral Load"

Bob then requests access to John Doe's viral load, which is also part of the non-sensitive medical data authorized by access control on condition that an inference is not triggered. The inference control process runs again as follows:

• Identify the inference channels in which the data to be consulted participates

→ IC2: Interferon, Viral Load, RBCs, T4/T8 Lymphocytes

• Determine if the inference channel (IC2) is active for John Doe

→ The Seropositivity is sensitive for John Doe, therefore IC2 is confirmed active

• Calculate the inference percentage based on the weight of the data and Bob's access history

→ Bob has accessed the treatment (Interferon) of channel IC2 and $I(IC2, Bob) = 100 * (0.35 + 0.5) = 85\%$

The first inference threshold (75%) is crossed and the obligation to warn the administrator of this risk of disclosure is required in accordance with the following rule:

```
<Policy PolicyId="ViralLoad:Access_modality">
  <Target>
  <!-- this policy concerns the reading of the Viral Load of
patient John Doe>
    <Resources>
      <Resource>
        <AttributeId>Data.Identity</AttributeId>
        <AttributeValue>Viral Load</AttributeValue>
      </Resource>
      <Resource>
        <AttributeId>Data.Owner</AttributeId>
        <AttributeValue>John Doe</AttributeValue>
      </Resource>
    </Resources>
    <Actions>
      <Action>
        <AttributeId>Action.Type</AttributeId>

        <AttributeValue>Read</AttributeValue>
      </Action>
    </Actions>
  </Target>
  <Rule RuleId="ViralLoad:Access_ViralLoad"
Effect="Permit">
    <Target>
      <Subjects>
        <Subject>
          <AttributeId>Subject.Role</AttributeId>
          <AttributeValue>Nurse</AttributeValue>
        </Subject>
      </Subjects>
    </Target>
    <VariableDefinition
VariableId="First_Inference_Threshold">75%</VariableDefini
tion>
    <Condition>
      <Apply FunctionId="function:double-greater-than-or-
equal">
        <AttributeDesignator AttributeId="InferenceExists"/>
        <VariableReference
VariableId="First_Inference_Threshold"/>
      </Apply>
    </Condition>
    <ObligationExpressions>
      <ObligationExpression
ObligationId="notifyAdministrator" FulfillOn="Permit">
        <AttributeAssignment AttributeId="text">
          <RessourceAttributeDesignator
AttributeId="ressource-id"/>
        </AttributeAssignment>
        <AttributeAssignment AttributeId="text">
          <AttributeValue>combined with other data previously
consulted can lead to the deduction of sensitive data
by</AttributeValue>
        </AttributeAssignment>
        <AttributeAssignment AttributeId="text">
          <SubjectAttributeDesignator AttributeId="subject-
id"/>
        </AttributeAssignment>
      </ObligationExpression>
    </ObligationExpressions>

  </Rule>
</Policy>
```

6. CONCLUSION AND PERSPECTIVES

As a result of this work, we succeeded in providing our privacy protection process with an inference control mechanism. Our security policy is thus capable of applying static access control rules, dynamically covering several aspects of the protection of privacy (anonymization, pseudonymisation, etc.), and preventing data deductions by inference. We proposed combining access and inference controls by specifying the inference limits to be respected, as well as the actions to be taken once these limits have been reached with regard to the obligations of the access control rules. We also explained how we converted the inferences into calculated measures so that they are comparable to the thresholds specified in the access control rules. The implementation of this mechanism has been provided by OLAP products, widely used in Business Intelligence (BI) information systems, and known for their ability to operate on complex and large databases. It is still important to mention that the effectiveness of our solution can only be guaranteed if all the inference channels of the application are identified. A lot of work is needed in the design phase to list all the functional dependencies between the data leading to these channels. The performance and optimization of processing and response time are among the areas we desire to explore in future works.

REFERENCES

[1] El Mokhtari, J., Abou El Kalam, A., Benhadou, S., Medroumi, H. (2019). PrivUML: A privacy metamodel. Procedia Computer System, 151: 53-60. https://doi.org/10.1016/j.procs.2019.04.011

[2] El Mokhtari, J., Abou El Kalam, A., Benhadou, S., Leroy, J.P. (2021). Transformation of PrivUML into XACML using QVT. Proceedings of the 12th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2020) In Advances in Intelligent Systems and Computing, Springer, 1383: 984-996. https://doi.org/10.1007/978-3-030-73689-7_93

[3] Barker, K., Askari, M., Banerjee, M., Ghazinour, K., Mackas, B., Majedi, M., Pun, S., Williams, A. (2009). A data privacy taxonomy. Lecture Notes in Computer Science, Springer-Verlag Berlin Heidelberg, 5588: 42-54. https://doi.org/10.1007/978-3-642-02843-4_7

[4] Ajam, N., Cuppens-Boulahia, N., Cuppens, F. (2010). Contextual privacy management in extended role based access control model. Lecture Notes in Computer Science, Springer Berlin Heidelberg, 5939: 121-135. https://doi.org/10.1007/978-3-642-11207-2_10

[5] Abou El Kalam, A., El Baida, R., Balbiani, P., Benferhat, S., Cuppens, F., Deswarte, Y., Miège, A., Saurel, C., Trouessin, G. (2003). Organization based access control. Proceedings POLICY 2003 - IEEE 4th International Workshop on Policies for Distributed Systems and Networks, pp. 120-131. https://doi.org/10.1109/POLICY.2003.1206966

[6] OASIS ― eXtensible Access Control Markup Language (XACML), Version 1.0, 2003. www.oasis-open.org›committees › oasis-xacml-1.0.pdf.

[7] El Mokhtari, J., Abou El Kalam, A., Benhadou, S., Leroy, J.P. (2021). Dynamic management of security policies in PrivOrBAC. International Journal of Advanced Computer Science and Applications (IJACSA), 12(6): 693-701. https://doi.org/10.14569/IJACSA.2021.0120681

[8] Andrieux, A., Czajkowski, K., Dan, A., Keahey, K., Ludwig, H., Nakata, T., Pruyne, J., Rofrano, J., Tuecke, S., Xu, M. (2011). Web services agreement specification (WS-Agreement). Global Grid Forum. http://www.ogf.org/documents/GFD.192.pdf.

[9] Biskup, J., Embley, D.W., Lochner, J.H. (2008). Reducing inference control to access control for normalized database schemas. Information Processing Letters, 106(1): 8-12. https://doi.org/10.1016/j.ipl.2007.09.007

[10] Biskup, J., Hartmann, S., Link, S., Lochner, J.H. (2010). Efficient inference control for open relational queries. International Federation for Information Processing in Lecture Notes in Computer Science, Springer Berlin Heidelberg, 6166: 162-176. https://doi.org/10.1007/978-3-642-13739-6_11

[11] Katos, V., Vrakas, D., Katsaros, P. (2011). A framework for access control with inference constraints. IEEE 35th Annual Computer Software and Applications Conference, pp. 289-297. https://doi.org/10.1109/COMPSAC.2011.45

[12] Paci, F., Zannone, N. (2015). Preventing information inference in access control. Proceedings of the 20th ACM Symposium on Access Control Models and Technologies (SACMAT'15), pp. 87-97. https://doi.org/10.1145/2752952.2752971

[13] Auxilia, M., Raja, K. (2016). Knowledge based security model for banking in cloud. Proceedings of the International Conference on Informatics and Analytics (ICIA-16), pp. 1-6. https://doi.org/10.1145/2980258.2980364

[14] Jebali, A., Sassi, S., Jemai, A. (2020). Secure data outsourcing in presence of the inference problem: Issues and directions. Journal of Information and Telecommunication, 5(1): 16-34. https://doi.org/10.1080/24751839.2020.1819633

[15] Suzuki, R., Suzuki, K., Morizumi, T., Hirotsugu, K. (2012). A hypergraph-based model against information leakage by inference. IEICE Transactions on Fundamentals of Electronics D, J95-D(4): 812-824.

[16] Sumiko, M., Kazuhiro, S., Tetsuya, M., Hirotsugu, K. (2014). Access control model for the My Number national identification program in Japan. IEEE 38th Annual International Computers, Software and Applications Conference Workshops, pp. 152-157. https://doi.org/10.1109/COMPSACW.2014.29

[17] Hirotsugu, K., Tetsuya, M. (2017). Access control model for the inference attacks with access histories. IEEE 41st Annual Computer Software and Applications Conference (COMPSAC), pp. 498-503. https://doi.org/10.1109/COMPSAC.2017.41