Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 00 (2023) 000–000

Procedia Computer Science

www.elsevier.com/locate/procedia

27th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2023)

Translating Usage Control Policies to Semantic Rules: A Model using OrBAC and SWRL

Nouha Laamech[a,*], Manuel Munier[a], Congduc Pham[b]

[a] *Universite de Pau et des Pays de l'Adour, E2S UPPA, LIUPPA, Mont-de-Marsan, France*
[b] *Universite de Pau et des Pays de l'Adour, E2S UPPA, LIUPPA, Pau, France*

Abstract

The increasing volume of data in various environments such as IoT and the need to maintain data privacy and security have led to the development of usage control models. Usage control policies are models that enable fine-grained access control over data by enforcing restrictions on how users can use the data. Semantic mechanisms, on the other hand, use context and meaning to identify potential security threats and prevent them from accessing sensitive information. Although not widely explored, merging these two techniques could create an efficient mechanism to help ensure the confidentiality, integrity, and availability of critical data and resources. This paper aims to encourage this research path by proposing a translation model that converts usage control rules into SWRL. In particular, we consider during our approach the notions of context, permission and prohibition. The proposition is validated by constructing a multi-layer proof of concept that use ontologies and OWL for implementing the translation model. Furthermore, to ascertain the practicality of our approach, a time processing evaluation is conducted, and the results are found to be satisfactory.

© 2023 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)
Peer-review under responsibility of the scientific committee of the KES International.

Keywords: Usage Control, Semantic Web Rule Language, Security;

1. Introduction

Nowadays, the distinctive focus on generating collaborations between smart devices to achieve efficient performance, as well as the requirement for third-party involvement, exposes devices, actors, and the entire connected system to potential security and safety issues. To address this, controlling and restricting access to resources is one of the fundamental mechanisms used to restoring order and ensuring transparency [1]. Therefore, access control models such as MAC [2], DAC [3] and RBAC [4] have emerged. These models are based on the establishment of a set of

* Corresponding author. Tel.: +33 5 58 51 37 00.
E-mail address: laamech.nouha@univ-pau.fr

1877-0509 © 2023 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)
Peer-review under responsibility of the scientific committee of the KES International.

access control rules called 'Permissions' in the form of subject, object, and action. Permissions define which subject or entity can access which object, and specify the actions that are allowed.

An anticipation of the need to introduce access and usage control mechanisms in the IoT and connected environments has been provided by various works in the literature such as [5] and [6], while also pointing out the challenges of introducing these mechanisms on distributed architectures and constrained devices. Moreover, the distributed nature of IoT networks leads to having several agents who do not share the same vocabulary nor the same roles, which can thus disturb the description of common usage control polices. Web semantics tackles this problem by guaranteeing interoperability and providing the ability for one system to interact with the units of another entity and exchange data with a shared and unambiguous meaning. This allows for knowledge recognition, computational reasoning, and the aggregation of different information systems. To leverage the benefits of the semantic web and enable reasoning deduction, one approach is to convert usage control rules into SWRL rules. SWRL (Semantic Web Rule Language) [7] is a rule language that enables the creation of semantic web rules using the Web Ontology Language (OWL). By expressing usage control policies as SWRL rules, it is possible to integrate them into a knowledge base that can be reasoned over using automated reasoning engines.

The authors in this article continue their research in the field of efficient data management within connected environments. During the analysis of the literature, it was noticed that despite the potential benefits of merging semantic and access control rules, this integration has not been widely adopted, and the benefits are not being fully realized [8]. For instance, a Semantic Policy Language was developed in [9], but this engages the creation of a new rule formalism based on XML that can be difficult to apprehend and adjust in already existing reasoning frameworks. Our proposal is to develop a translation model that can convert Organization Based Access Control (OrBAC) [10] rules to SWRL rules. SWRL is a well known rule language that is used to describe rules in the Semantic Web, while OrBAC is a model that is used to define usage control policies.

The rest of the paper is organized as follows. In section 2, a motivating scenario is presented that will be used through the rest of the paper. Background information are discussed in section 3 about existing semantic and access control security-preserving solutions. Section 4 describes the OrBAC model, the IdSM ontology, and the motive behind those choices. Section 5 explores in details the translation model process, while section 6 presents experimental evaluation. Finally, we conclude with a discussion and some future work.

2. Motivating scenario

We consider a connected environment that involves multiple agents who need to share data with one another. The agents could be individuals or organizations. For example, in smart agriculture, precision farming involves using data from sensors, drones, and other sources to create detailed maps of fields, showing variations in soil moisture, nutrient levels, and other factors. This data can be shared with state services and other stakeholders to develop more precise and efficient public support practices such as financial aid estimation or anti-waste procedures examination. Additionally, farmers can use available data to apply fertilizers and pesticides only where they are needed, reducing waste and minimizing environmental impact. By sharing data and collaborating with experts in the agriculture industry, farmers can improve their yields, reduce costs, and achieve greater sustainability in their operations.

Although this basic architecture is widely present in current ecosystems, there are still issues to be resolved. At this point, one of the primary challenges is ensuring information security while allowing authorized access to data. This requires a comprehensive security strategy, including access controls, encryption [11], and monitoring [12]. However, it is crucial that a usage control express rules that not only grant authorization but also impose obligations and prohibitions. Moreover, the usage control model must take in consideration the cultural and organizational differences. Different organizations may have different data sharing practices, policies, and governance structures, which can create friction when trying to establish data sharing policies. Furthermore, taking context awareness into account is essential to ensure dynamic policies. Additionally, incorporating context awareness can increase efficiency by giving the opportunity for data consumers to assess the quality of a shared data and therefore grand them visibility on which data is the most relevant regarding their goals. All these issues are taken into account in the Organization Based Access Control (OrBAC) model. Another challenge is the ability for different systems and applications to work together and exchange data seamlessly. Achieving interoperability requires standardization of data formats, protocols, and interfaces. Many

ontologies were developed to tackle this problem and to describe as many various domains as possible, either by the reuse of available ontologies [13][14][15], the alignment of ontologies [16], or ontology learning [17].

Considering all of the above, the following research questions emerge: (i) how to exploit usage control models rather than traditional access control ones in order to still take into account the usage of resources even after granting the access (ii) how to exploit the full potential of the semantic web to allow more efficient reasoning processes and better access control expressiveness (iii) how to consider additional contextual factors, such as the data quality, the value of an observation, or the location from which they are accessing the resource. Furthermore, permissions should not be the only rule relied on to express control policies, but obligations and prohibitions as well. In the following section, we describe our proposal which addresses all the three above-mentioned issues by building a conversion model from OrBAC rules to SWRL.

3. Related work

Our contribution seeks to complement existing solutions that promote the adoption of access control and semantic mechanisms to enhance security. These solutions can be broadly categorized into two types: those that focus on access control or semantic mechanisms individually and those that combine both approaches for a comprehensive security solution.

3.1. Access control and Semantic-based mechanisms: individual approaches

In [18], authors use ABE-enabled ABAC, an attribute-based model taking paradigms from ABAC models and architectures to introduce attribute-based cryptographic elements for data privacy. They designed a conceptual architecture that use encryption and key generation algorithm to feed attribute and policy responses from the PDP when a data consumption or data encryption request is submitted. Authors finally discuss the translation restrictions of ABAC access policies to useful parameters in the encryption and decryption algorithm of ABE.

In [19], an attribute-based secure access control mechanism (SACM) is proposed for IoT-Health utilizing the federated deep learning (FDL), where they manage to discover, using graph convolutional networks, the relationship between users' social attributes and their trusts. The authors evaluate their framework in IoT-enabled healthcare tested and demonstrate its effectiveness.

Several semantic models are defined to enable interoperability between solutions from different providers and among various activity sectors and environments [20]. Authors in [21] develop a privacy-preserving SRSE scheme for the cloud environments using semantic-aware ranked search. They adopt the Latent Dirichlet Allocation topic model to generate the topic-based semantic information-embedded vectors for documents and queried keywords using k-means clustering algorithm. On the other hand, authors in [22] propose a multi-label semantics preserving based deep cross-modal hashing (MLSPH) method. MLSPH firstly calculate semantic similarity of the original data, and then introduce a memory bank mechanism to preserve the multiple labels semantic similarity constraints and enforce the distinctiveness of learned hash representations over the training batch. They experimented their model using several benchmark datasets and obtained successful results.

3.2. Access control and Semantic-based mechanisms: combined approaches

Semantic Attribute-Based Access Control (SABAC) has emerged that takes into account the semantics of attributes by combining ABAC with semantic technologies. Arshad et al. [23] provides a comprehensive summary of the conducted research studies that is useful for researchers who want to enter and make contributions to this research area. Furthermore, the paper identifies open problems and possible research entry points by demonstrating the advantages and the challenges of this approach. Choi et al. proposes Onto-ACM (ontology-based access control model) [24], a semantic analysis model that address the difference in the permitted access control between service providers and users. The proposed model is a model of intelligent context-aware access for proactively applying the access level of resource access based on ontology reasoning and semantic analysis method.

In [25], authors propose techniques for information security and privacy protection for Smart Spaces based on the Smart-M3 platform. They build a security framework that consider a context-aware role-based access control scheme.

The modeling of the access control scheme is done with ontological techniques and OWL, while the rules are set using CLIPS. An evaluation is proceeded to measure the response time for the access requests and prove the efficiency of the model.
Based on the semantic integration nature of XML data, [26] proposes a data access control model for users in big data contexts. A global visual range of inverted XML structure is built through the semantic dependency between data and the integration process. Evaluations shows good results and has high access efficiency.

4. Modelization process

4.1. The Organization Based Access Control Model

We have chosen to illustrate our approach using OrBAC as it is considered one of the most expressive access control models available. However, it is important to note that our approach can also be applied to other access control models.

In OrBAC, subject, action and object are concrete entities that are abstracted respectively into role, activity, and view within an Organization. For instance, the relationship `Empower(org,r,s)` means that the organization `org` employs subject `s` in a role `r`, `Consider(org,activity,action)` means that `activity` employs `action`, and `Use(org,v,o)` implies that the view `v` employs the object `o`. Thus, instead of modeling the policy using the concepts of subject, action and object related to the implementation, the OrBAC model proposes to reason with the roles, activity and view that are assigned in the organization.

OrBAC manage to automatically derive concrete rules from abstract rules using a security policy that is defined within an organization. This policy is a set of rules (permissions, prohibitions, or obligations). These security rules are not static in the sense that their application depends on the context, which is a concept that was explicitly introduced in OrBAC. A context is a special condition between the subject, the object and the action linked by the Hold concept that controls the activation of the rules expressed in the access control policy. For example, suppose that the organization University defines in its security policy the following abstract rule: "the role professor is allowed to perform the activity consult on the view scientific article at context academic-year". Suppose now that the subject Bob plays the role professor, the action select implements the activity consult, the object conference_paper.doc is used in the view scientific article, and the current context is academic-year for the University. Then the concrete rule which can be derived from the previous abstract rule is: Bob is allowed to select conference_paper.doc file.

For the previously mentioned example, the permission will be expressed as follows: *Permission('University','researcher','access','conf_article','academic-year')*. In the same manner, to express prohibition we use *Prohibition('org','r','a','v','context')* and *Obligation('org','r','a','v','context','vcontext')* with 'context' being the context activation, and 'vcontext' the context violation.

4.2. Semantic description using ontologies

In semantic web, an ontology is the result of an exhaustive and rigorous formulation of a domain conceptualization. It's a set of entities linked with eachother using object and data properties. IdSM-O (IoT data sharing management ontology) [27] is an ontology that tackles the scenario mentioned in the previous section by aligning OrBAC, SNN and Prov-O ontologies into a new one and extends the result by adding data sharing description modules. We extend the already existing model by adding new components to describe contexts, while remaining compliant with existing standards. Indeed, we add the `Hold` concept already introduced in the OrBAC model to link a subject, action, and object with a given context. Figure 1 shows the added features in regard of the already exiting IdSM modules.

The extensions are related to:

- Concepts: we add a class `Context`, to describe the context of a certain operation that depends on the domain. To instantiate this entity, we create the class `Hold` that have the role to link the subject, action, and object with the entity `Context` and create rules in case those contexts are activated. We also add the concept `Operation` that link an `Action`, `Object` and `Subject`. In the case of data sharing management, `Operation` represents a request for a certain action on a view.

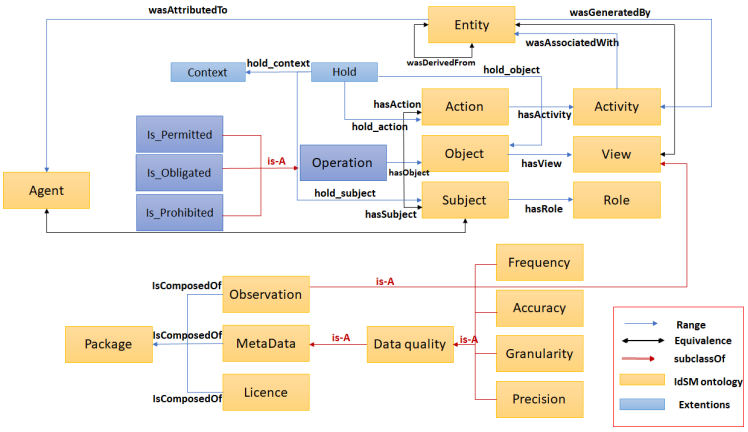

Fig. 1: IdSM overview

- Object properties: we use relations to link the different concepts with each other. Therefore, a `Hold` instance is linked with the object, subject, action, and context using respectively `hold_object`, `hold_subject`, `hold_action`, and `hold_context`.

To understand Fig. 1, let's consider the following example. We define Sam as an individual of the orbac:Subject class who has a weather sensor for his wheat field, which are respectively instances of the sosa:Sensor and orbac:View classes. This sensor has two outputs, humidity and temperature. A usage preference rule is defined for the two outputs of the device. The first rule grants full permission on the temperature observations of Sam's wheat field to all other farmers. The second rule gives permission to access the moisture content of his land only for consultation, and only by people with the role of "researchers". Alice and Bob are instances of orbac:Subject class who request to consult both views (temperature and humidity) . Alice has the role of nursery worker, which is a sub role of farmer (instance of orbac:Role). Bob has the role researcher. By expressing those requests and Sam's usage preference using OrBAC rules, and by translating those rules to SWRL, rule engine can be used to deduce that Alice request to access the temperature observation and Bob request to consult the humidity observation is granted. Therefore they can access the instances of the idsm:Package class.

5. Proposed model: translation from access control rules to SWRL

To translate access control rules into SWRL rules, we go through three main steps: writing the rule using the access control model, retrieval of the main information using JSON, and the building of the SWRL rules. The frames below describe these steps.

5.1. Non-contextual rules

Traditional access control rules do not take in consideration contexts nor prohibition or obligation. Therefore, the policy rule format that we fall back to for most of access control models is :

R1 ≡ Permission(⟨**the_role**⟩,⟨**the_activity**⟩,⟨**the_view**⟩)

Predicate logic is a formal system that uses predicates and quantifiers to express relationships between objects, properties, and concepts. The rule R1 can be interpreted in predicate logic as follows:

$\forall$ s, $\forall$ o, $\forall\alpha$, $\forall$ r, $\forall$ v, $\forall$ a Permission(r, a, v) $\wedge$ empower(r, s) $\wedge$ use(v, o) $\wedge$ consider(a, α) $\Rightarrow$ is_Permitted(s, α, o)
To translate this into SWRL rules, we derive the needed information using a JSON structure.

Rule = ”permission”
Role = ”**the_role**”
Activity = ”**the_activity**”
View = ”**the_view**”
Name_context = ” ”
Expression_context = ” ”

Finally, we build the final SWRL rule interpretable by reasoners. In our translating model, we incorporate the concept of inheritance to efficiently manage permissions and restrictions for role, object, and action concepts. By leveraging inheritance, we can establish hierarchical relationships between these concepts, allowing permissions and restrictions to be inherited by sub-roles, sub-objects, and sub-actions. This minimizes the need for duplicative or redundant access control rules, simplifying the management of access control policies and reducing the risk of errors or inconsistencies. This model remains unchanged and therefor applicable for all domains, except for the content highlighted in bold, which varies based on the information extracted from the access control rule.

idsm:Operation(?op)

$\wedge$ idsm:hasSubject(?op, ?subject) $\wedge$ orbac:empower(?subrole, ?subject)
$\wedge$ orbac:role_subrole(?role, ?subrole) $\wedge$ orbac:Role(?role) $\wedge$ sameAs(?role,$\langle$**the_role**$\rangle$)

$\wedge$ idsm:hasAction(?op, ?action) $\wedge$ orbac:consider(?subactivity, ?action)
$\wedge$ orbac:activity_subactivity(?activity,?subactivity) $\wedge$ sameAs(?activity,$\langle$**the_activity**$\rangle$)

$\wedge$ idsm:hasObject(?op, ?object) $\wedge$ orbac:use(?subview, ?object)
$\wedge$ orbac:view_subview(?view,?subview) $\wedge$ sameAs(?view,$\langle$**the_view**$\rangle$)

$\rightarrow$ idsm:is_Permitted(?op)

Operation refers to a request made by a certain subject, to do a certain action on a certain object. An instance of operation will represent the bridge between a given domain and access control features ’subject’, ’action’ and ’object’, using respectively the relations ’hasSubject’, ’hasAction’ and ’hasObject’. In the same way, ’subject’, ’action’ and ’object’ are respectively linked with ’roles’, ’activity’ and ’view’ using the object properties ’empower’, ’consider’, and ’use’.

5.2. Context aware rules for access control policies

Some access control policies add contextual features to existing BAC models. For instance, OrBAC use contexts to provide a more adaptive and dynamic approach, which allows it to evolve into a usage control model that goes beyond traditional access control. Furthermore, OrBAC enables more expressiveness by considering obligations and prohibitions in addition to permissions. Therefore, we opted to construct our translation model based on it since it’s one of the access control models with the most extensive coverage.

The logical process is carried out in three main steps.

A- The definition of an OrBAC rule: this rule is similar to the one we previously saw in Section 4.1, but with the addition of context. By considering contextual factors that may impact the rule’s application, we can ensure that it is enforced appropriately and consistently. Incorporating context into rule design and enforcement can help to improve overall system performance and ensure that rules are tailored to specific situations. This can help to prevent errors and inconsistencies, as well as ensure that resources are used efficiently and effectively.

R1 ≡ **Permission**(**⟨no_org⟩**, **⟨the_role⟩**, **⟨the_activity⟩**, **⟨the_view⟩**, **⟨the_context⟩**)

B- Information retrieval and storage in JSON: information retrieval can be effectively implemented using JSON, which is a lightweight data interchange format that is easily readable and writable by both humans and machines. We retrieve the rule type (permission, obligation, or prohibition), OrBAC attributes, and the context expression that will trigger a certain policy. This context expression can be related to a location, time, or physical surrounding.

Rule = "**permission**"
Role = "**the_role**"
Activity = "**the_activity**"
View = "**the_view**"
Name_context = "**the_context**"
Expression_context = "**logical_conjunction** "

C- Context treatment: we first proceed to express under which circumstance the context is considered to be activate. This expresion is dynamic and will depend on each domain, user, and ontology entities :

idsm:Operation(?op)

∧ idsm:hasSubject(?op, ?subject) ∧ idsm:hasAction(?op, ?action) ∧ idsm:hasObject(?op, ?object)

∧ **logical_conjunction**

∧ swrlx:makeOWLThing(?h,?op)

→ orbac:Hold(?h) ∧ orbac:hold_subject(?h, ?subject)∧ orbac:hold_action(?h, ?action) ∧ orbac:hold_object(?h, ?object) ∧ orbac:hold_context(?h,**⟨the_context⟩**)

The final step is to build an SWRL rule to grant permission in case the context is activated.

Let's apply this on our scenario. Let's assume a data provider is a farmer that wants his water_meter observation to be accessible for all farmers as long as they're located in Europe. The OrBAC rule regarding this policy and the extracted information are:

R1 ≡ $Permission('Farm','farmer','access','water\_meter','Europe\_context')$

Role = "**farmer**"
Activity = "**access**"
View = "**water_meter**"
Name_context = "**Europe_context**"
Expression_context = "**hasLocation(?s,'Europe')**"

Using the IdSM ontology and the previously presented steps, the SWRL rule derivation to express context activation is as follows:

idsm:Operation(?op) ∧ idsm:hasSubject(?op,?subject) ∧ idsm:hasAction(?op,?action) ∧ idsm:hasObject(?op,?object) ∧ **idsm:hasLocation(?subject, "Europe")** ∧ swrlx:makeOWLThing(?h,?op)

idsm:Operation(?op)

∧ idsm:hasSubject(?op, ?subject) ∧ orbac:empower(?subrole, ?subject)
∧ orbac:role_subrole(?role, ?subrole) ∧ orbac:Role(?role) ∧ sameAs(?role,**⟨the_role⟩**)

∧ idsm:hasAction(?op, ?action) ∧ orbac:consider(?subactivity, ?action)
∧ activity_subactivity(?activity,?subactivity) ∧ sameAs(?activity,**⟨the_activity⟩**)

∧ idsm:hasObject(?op, ?object) ∧ orbac:use(?subview, ?object)
∧ view_subview(?view,?subview) ∧ sameAs(?view,**⟨the_view⟩**)

∧ orbac:Hold(?h) ∧ orbac:hold_subject(?h, ?subject) ∧ orbac:hold_action(?h, ?action)
∧ orbac:hold_object(?h, ?object) ∧ orbac:hold_context(?h, **⟨the_context⟩**)

→ idsm:is_Permitted(?op)

Fig. 2: The rule that grants permission as the final step

→ orbac:Hold(?h) ∧ orbac:hold_subject(?h, ?subject) ∧ orbac:hold_action(?h, ?action) ∧ orbac:hold_object(?h, ?object) ∧ orbac:hold_context(?h,"**Europe_context**");

The final rule to grand permission in case of a successful match is:

Idsm:Operation(?op) ∧ idsm:hasSubject(?op, ?subject) ∧ orbac:empower(?subrole, ?subject) ∧ orbac:role_subrole(?role, ?subrole) ∧ orbac:Role(?role) ∧ sameAs(?role,"**farmer**") ∧ idsm:hasAction(?op, ?action) ∧ orbac:consider(?subactivity, ?action) ∧ activity_subactivity(?activity,?subactivity) ∧ sameAs(?activity,"**access**") ∧ idsm:hasObject(?op, ?object) ∧ orbac:use(?subview, ?object) ∧ view_subview(?view,?subview) ∧ sameAs(?view,"**water_meter**") ∧ orbac:Hold(?h) ∧ orbac:hold_subject(?h, ?subject) ∧ orbac:hold_action(?h, ?action) ∧ orbac:hold_object(?h, ?object) ∧ orbac:hold_context(?h, "**Europe_context**")
→ is_Permitted(?op))

To apply the same permission model to prohibited policies, we simply need to change the rule type from "permission" to "prohibited". After making this modification, we can apply the same approach that was used for permission policies. An example of a prohibition rule would be for a farmer to not want his observations shared with anyone during drought periods. This could be implemented as a "prohibited" rule in the permission model, where the operation that represents this subject, action, and object is denied during the specified period using `is_Prohibited`. On the other hand, obligation rule refers to the requirement for users or entities to comply with certain rules, policies, or regulations in a certain context. The rule takes into account ⟨context_violation⟩ which is an additional condition that make a particular subject violates the defined context constraint. Although developing a translation model for obligation rules is achievable and does not require significant deviation from what we did with permission and prohibited policies, we chose not to pursue it in this paper. Our main focus is to proof that we're able to convert OrBAC to SWRL, we therefore consider that obligations is an added level of expressiveness that is not mandatory at this stage of developement but will be adressed in future work (see Section 7 for perspectives).

6. Proof of concept and experimental evaluation

Based on the model defined in the previous section, we created a proof of concept to automatize the three main steps for translation rules. As shown in Fig. 3, this model consists of three layers: an ontology, the domain-specific vocabulary, and the generation of SWRL rules from usage control rules. The initial step involves providing an

ontology (OWL file) that describes a specific domain as input. Subsequently, the vocabulary layer is in charge of providing instances of concepts and relations defined in the ontology. Lastly, the translation process takes place in the final layer. We developed this model using Java and JavaFX for vocabulary and access control rules instantiation.

Due to its focus on data management, IdSM-O already addresses access control considerations and can be applied to a broad array of fields. As such, we employ it as the standard input for the first layer. Nevertheless, since this proof of concept can manage different imports with no issues, we can anticipate the inclusion of additional ontologies that describe a certain vocabulary in a specific domain. This auxiliary ontology would then be added as an import, and its specific concepts can be used in addition to IdSM-O access control modules. This expansion of ontological resources enhances the versatility and adaptability of the system to meet domain-specific requirements.

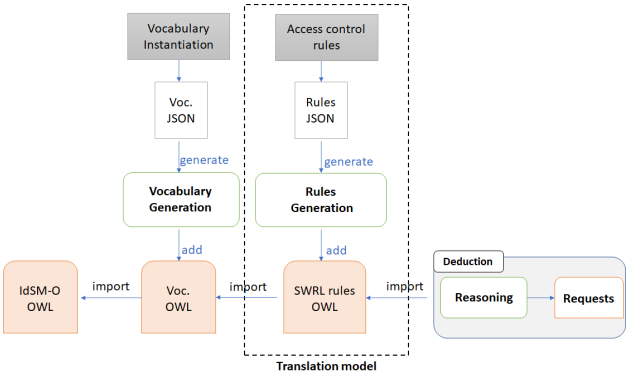

Fig. 3: Sequencing of the different modules

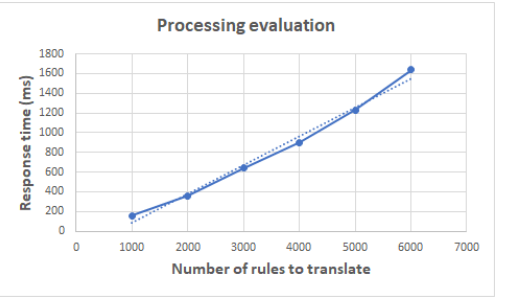

Fig. 4: Evaluation of the translating model

We conduct some experiments to measure the performance of this proof of concept. We evaluate the response time required to translate a number of rules. The response time is measured from the moment the ontology is loaded to the moment the generated SWRL rules are added to a final OWL file that contains the ontology, vocabulary, and rules, ready for inference reasoning and deductions.
We observe that this proof of concept can support a large number of translation requests within reasonable processing time. Indeed, the response time does not exceed 2 seconds even for 6000 rules.

7. Conclusion and perspectives

By integrating Semantic Web technologies with access control mechanisms, it becomes possible to apply access control policies dynamically based on the context of the resource being accessed. In this paper, our main contribution is the translation process of OrBAC rules into SWRL rules in order to allow the definition of more expressive access control policies. The main benefits of this work are:

- consideration of contexts during the translation process,
- flexibility of the model that allows it to be applied in a wide range of domains,
- incorporation of inheritance for more concise and modular policy definitions,
- implementation of a three-layer proof of concept to manipulate the model translation.

In terms of perspectives, several current limitations must be studied in future works:

- in the context of rule conflict avoidance, it is necessary to establish rule priorities at an abstract level in order to effectively manage conflicts that may arise between different rules. This involves defining a hierarchy of rules based on their importance or relevance to the specific problem or domain being addressed,
- consideration of additional parameters such as organizations and obligations rules during the translation model,
- possibility to include other ontologies in addition to IdSM as the input for the first layer.

Acknowledgments

This work is supported by the Conseil Départemental des Landes (PhD grant to N.Laamech).

References

[1] Shantanu Pal, Ali Dorri, and Raja Jurdak. Blockchain for iot access control: Recent trends and future research directions. *Journal of Network and Computer Applications*, 203:103371, 04 2022.

[2] Hakan Lindqvist. Mandatory access control. *Master's thesis in computing science, Umea University, Department of Computing Science, SE-901*, 87, 2006.

[3] Mumina Uddin, Shareeful Islam, and Ameer Al-Nemrat. A dynamic access control model using authorising workflow and task-role-based access control. *IEEE Access*, 7:166676–166689, 2019.

[4] Georgios Fragkos, Jay Johnson, and Eirini Eleni Tsiropoulou. Dynamic role-based access control policy for smart grid applications: An offline deep reinforcement learning approach. *IEEE Transactions on Human-Machine Systems*, 52(4):761–773, 2022.

[5] Shantanu Pal, Ali Dorri, and Raja Jurdak. Blockchain for iot access control: Recent trends and future research directions. *Journal of Network and Computer Applications*, 203:103371, 04 2022.

[6] Denisse Muñante Arzapalo, Vanea Chiprianov, Laurent Gallon, and Philippe Aniorté. A model-driven security requirements approach to deduce security policies based on orbac. In Dongdai Lin, Moti Yung, and Jianying Zhou, editors, *Information Security and Cryptology*, pages 150–169, Cham, 2015. Springer International Publishing.

[7] Ian Horrocks, Peter F Patel-Schneider, Harold Boley, Said Tabet, Benjamin Grosof, Mike Dean, et al. Swrl: A semantic web rule language combining owl and ruleml. *W3C Member submission*, 21(79):1–31, 2004.

[8] Nouha Laamech, Manuel Munier, and Congduc Pham. Towards a data provenance model for private data sharing management in iot. In *2021 IEEE 25th International Enterprise Distributed Object Computing Workshop (EDOCW)*, pages 210–215, 2021.

[9] Ines Akaichi and S. Kirrane. A semantic policy language for usage control. In *International Conference on Semantic Systems*, 2022.

[10] Cuppens Frédéric and Mième Alexandre. Or-bac, organization based access control. *The Review of Politics, Journées Druide*, 2004.

[11] Faiza Loukil, Chirine Ghedira-Guegan, Khouloud Boukadi, and Aïcha-Nabila Benharkat. Privacy-preserving iot data aggregation based on blockchain and homomorphic encryption. *Sensors*, 21(7):2452, 2021.

[12] Kallol Karmakar, Vijay Varadharajan, Pete Speirs, Michael Hitchens, and Aron Robertson. Sdpm: A secure smart device provisioning and monitoring service architecture for smart network infrastructure. *IEEE Internet of Things Journal*, PP:1–1, 12 2022.

[13] Salma Sassi, Anis Tissaoui, and Richard Chbeir. Leonto+: a scalable ontology enrichment approach. *World Wide Web*, 25, 02 2022.

[14] Piotr Sowiński, Katarzyna Wasielewska-Michniewska, Maria Ganzha, Marcin Paprzycki, and Costin Badica. *Ontology Reuse: The Real Test of Ontological Design*. 09 2022.

[15] Alfonso Castro, Victor Villagra, Paula Garcia, Diego Rivera, and David Toledo. An ontological-based model to data governance for big data. *IEEE Access*, 9:109943–109959, 01 2021.

[16] Yuan He, Jiaoyan Chen, Denvar Antonyrajah, and Ian Horrocks. Bertmap: A bert-based ontology alignment system. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36:5684–5691, 06 2022.

[17] Chuangtao Ma and Bálint Molnár. Ontology learning from relational database: Opportunities for semantic information integration. *Vietnam Journal of Computer Science*, 9:31–57, 02 2022.

[18] Alex Chiquito, Ulf Bodin, and Olov Schelén. Attribute-based approaches for secure data sharing in industrial contexts. *IEEE Access*, PP:1–1, 01 2023.

[19] Hui Lin, Kuljeet Kaur, Xiaoding Wang, Georges Kaddoum, Jia Hu, and Mohammad Hassan. Privacy-aware access control in iot-enabled healthcare: A federated deep learning approach. *IEEE Internet of Things Journal*, PP:1–1, 09 2021.

[20] Sabrina Kirrane, Serena Villata, and Mathieu d'Aquin. Privacy, security and policies: A review of problems and solutions with semantic web technologies. *Semantic Web*, 9:1–10, 01 2018.

[21] Qian Zhou, Hua Dai, Zheng Hu, Yuanlong Liu, and Geng Yang. Accuracy-first and efficiency-first privacy-preserving semantic-aware ranked searches in the cloud. *International Journal of Intelligent Systems*, 37, 08 2022.

[22] Xitao Zou, Xinzhi Wang, Erwin Bakker, and Song Wu. Multi-label semantics preserving based deep cross-modal hashing. *Signal Processing: Image Communication*, 93:116131, 01 2021.

[23] Hamed Arshad, Christian Johansen, and Olaf Owe. Semantic attribute-based access control: A review on current status and future perspectives. *Journal of Systems Architecture*, 129:102625, 06 2022.

[24] Chang Choi, Junho Choi, and Pankoo Kim. Ontology-based access control model for security policy reasoning in cloud computing. *The Journal of Supercomputing*, 67, 03 2014.

[25] Shohreh Hosseinzadeh, Seppo Virtanen, Natalia Diaz Rodriguez, and Johan Lilius. A semantic security framework and context-aware role-based access control ontology for smart spaces. pages 1–6, 07 2016.

[26] M. Wang, J. Wang, Lihong Guo, and L. Harn. Inverted xml access control model based on ontology semantic dependency. *Computers, Materials and Continua*, 55:465–482, 01 2018.

[27] Nouha Laamech, Manuel Munier, and Congduc Pham. Idsm-o: An iot data sharing management ontology for data governance. In *Proceedings of the 14th International Conference on Management of Digital EcoSystems*, MEDES '22, page 88–95, New York, NY, USA, 2022. Association for Computing Machinery.