SBAC: "A Semantic–Based Access Control Model"

S. Javanmardi, M. Amini, R. Jalili, and Y. Ganjisaffar

Network Security Center, Computer Engineering Department,
Sharif University Of Technology, Tehran, Iran
{s_javanmardi,m_amini,ganji}@ce.sharif.edu, jalili@sharif.edu

Abstract. Semantic Web is the vision for future of current Web which aims at automation, integration and reuse of data among different Web applications. The shift to Semantic Web applications poses new requirements for security mechanisms especially in the access control models as a critical component of security systems. Access to resources can not be controlled in a safe way unless the access decision takes into account the semantic relationships among entities in the data model under the Semantic Web. Decision making for granting or revoking access requests by assuming entities in isolation and not considering their interrelations may result in security violations. In this paper, we present a Semantic Based Access Control model (SBAC) which considers semantic relations among different entities in the decision making process. For accurate decision making, SBAC considers semantic relations among entities in all domains of access control, namely the subject domain, the object domain and the action domain. To facilitate the propagation of policies in these three domains, we show how different semantic interrelations can be reduced to the subsumption problem. This reduction enhances the space and time complexity of the access control mechanisms which are based on SBAC.

1 Introduction

Semantic Web is the extension of current Web which gives information a well-defined meaning, makes machines capable of interpreting and processing the information. The shift from current Web to semantic aware environments such as Semantic Web poses new security requirements [1, 2] especially in the field of access control. Access control is a mechanism that allows owners of resources define, manage and enforce access conditions applicable to each resource [3]. A semantic aware access control mechanism should assure that only eligible users are authorized to be granted an access right and each eligible user must be able to access all the resources that s/he is authorized for [4]. Traditional access control models like MAC, DAC and RBAC fail to address this issues since they do not consider the rich semantic relations in the data model under Semantic Web [5]. In other words, decision making based on isolated entities while ignoring the semantic interrelationships among them may result in *illegal inferences* by

unauthorized users and *incomplete granting of access rights*. Fig. 1 shows a part of Bank-Service ontology. The ovals show concepts and individuals and labels on the directed arcs shows axioms and properties. Individuals are represented by 'Is_A' label on the arcs. For example, consider the concept 'Credit Card' which is union of 'Master Card' and 'Visa Card', if a user is eligible to know about the latest transactions on credit cards issued by the bank while s/he is prevented from accessing the same information for visa cards, then s/he can guess some information about them which is illegal. On the other hand, when a bank authority needs to know some information about the concept 'Letter of Credit' for some decision making then s/he should be authorized for reading the information about an equal concept like 'Documentary Credit' for more accurate decision making.

To overcome these challenges, there is a need for semantic aware access control systems consistent with the semantic data model under the Semantic Web. In this paper, we present a Semantic Based Access Control model (SBAC) that authenticate users based on the credentials they offer while requesting an access right. Ontologies are used for modelling the entities along with their semantic interrelations in three domains of access control, namely subject domain, object domain and action domain. Decision making in SBAC for granting or denying an access request is automated by inference processes according to the semantic relation among entities. Based on the OWL [6] ontology language, we show how semantic interrelations can be effective in the authorization process; and for enhancing the expressiveness of authorization rules defined in SBAC, we show how rule languages like SWRL [7] can be applied. Since a general semantic relation called *subsumption* can facilitate the policy propagation, in SBAC we try to reduce different semantic interrelations to the subsumption problem.

The remainder of this paper is as follows: Section 2 describes the related works on this topic and section 3 states fundamentals of SBAC. Semantic authorization flow of access rights in different levels of an ontology are described in section 4. In section 5, the formal definition of SBAC is presented and it is shown how the reasoning can be done in different domains of access control. Finally, section 6 underlines some concluding regards and future research lines.

2 Related Work

Access control systems for protecting Web resources along with credential based approaches for authenticating users have been studied in recent years [3]. With the advent of Semantic Web, new security challenges were imposed to security systems. Bonatti et al in [2] have discussed open issues in the area of policy for Semantic Web community such as important requirements for access control policies. Developing security annotations to describe security requirements and capabilities of web services providers and requesting agents have been addressed in [8].

Object-Oriented authorization models for databases were the first models that tried to consider the semantic relationships for authorization. Such models

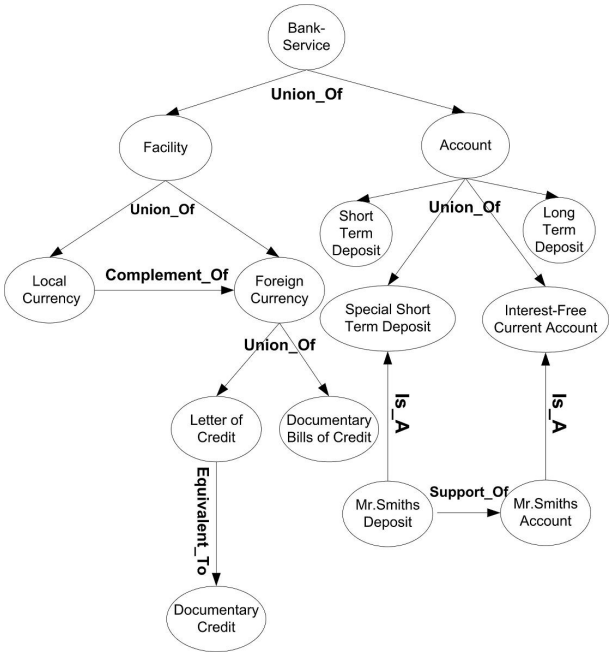

Fig. 1. A Part of Bank-Service Ontology

showed the effect of the semantic relationship like subclass/superclass in access decision making [9]. File–level access control systems were studied in [10] for protecting HTML resources. In the next layer, there are XML based approaches such as XACML (eXtensible Access Control Markup Language) [11] and XR-BAC (XML Role-Based Access Control) [12] that have attempted to express policies for controlling accesses to XML resources. Finin et al have proposed policy languages like Rei [13] based on Semantic Web languages like RDF and DAML+OIL and have developed a framework, Rein, based on Rei. In the ontology layer, Qin et. al. [4] proposed a concept level access control model which considers some semantic relationships in the level of concepts in the objects domain. In this paper, we present SBAC as an access control model based on OWL ontology language that considers semantic relationships in different levels of an ontology (Concept, Property, Individual) and in all the domains of access control (Subject, Object, Action). For enhancing the expressiveness and inference abilities, SBAC uses SWRL, a Horn clause rule extension to OWL. The derivation rules are in the form of $antecedent \Rightarrow consequent$ where both antecedent and consequent are conjunctive consequences of atoms which can be concepts, individuals or properties [7]. It seems that the future access control systems will be proof based systems like client based access control systems which work with undecidable but more expressive logics than Description Logic which is under OWL.

3 Introduction to SBAC

Like most of the other access control systems, SBAC makes its decisions on three domains: Subject, Object and Action. Fundamentally, SBAC consists of three basic components: Ontology Base, Authorization Base and Operations. Ontology Base is a set of ontologies: Subject-Ontology (SO), Object-Ontology (OO) and Action-Ontology (AO). These ontologies are describes in the following:

OO : is an Object Ontology for describing objects. Objects are entities which are accessed and/or modified. An object belongs to an Object-Ontology which shows the structure in which the objects (Concepts, Individuals and Properties) are organized along with the semantic relationships among them. Fig. 1 is an example of OO.

SO : is the Subject Ontology where subjects are active entities which require access to objects. Subjects are concepts or individuals in Subject-Ontology, Fig. 2.a) shows a Subject-Ontology which is based on credentials. Presenting credentials determine users eligibility for accessing a resource.

AO : Actions depend on the type of the actions that subjects aim to execute on an object. Each action type is a concept in the ontology and the actions are individuals of the concept defined in AO. Fig. 2.b) demonstrates an example of Action Ontology.

By modeling the access control domains using ontologies, SBAC aims at considering semantic relationships in different levels of an ontology to perform inferences to make decision about an access request. Authorization Base is a set of authorization rules in the form of $(s, o, \pm a)$ in which s is an entity in SO , o is an entity defined in OO, and a is an action defined in AO. In other words, a rule determines whether a subject which presents a credential s can have the access right a on object o or not. Predefined access rights can be saved in Authorization Base in the form of authorization rules and for making decision for incoming requests (grant/deny), inference is done based on the semantic relationships between the requested authorization and the explicit authorization rules in Authorization Base. In fact, inferences on the explicit authorization rules result in some implicit authorizations rules. For example, if an explicit authorization rule states that a subject can read the URI, http://Banskservices/Account, then if s/he requests for an access right to read a sub–object like http://Banskservices/Accoumt/ShortTermDeposit, then the latter can be inferred from the former without saving its authorization rule explicitly. Since SBAC works based on inference, for preventing propagation of same decision (grant/deny) on all the inferred rules, it allows the definition of exception rules with higher priority. For example, an exception rule can be defined if the authority of a bank wants to prohibit the credit cards issued from a specific bank from settling money to any account in $Bank_x$ while there is another explicit authorization rule that lets all credit cards settle money in any account.

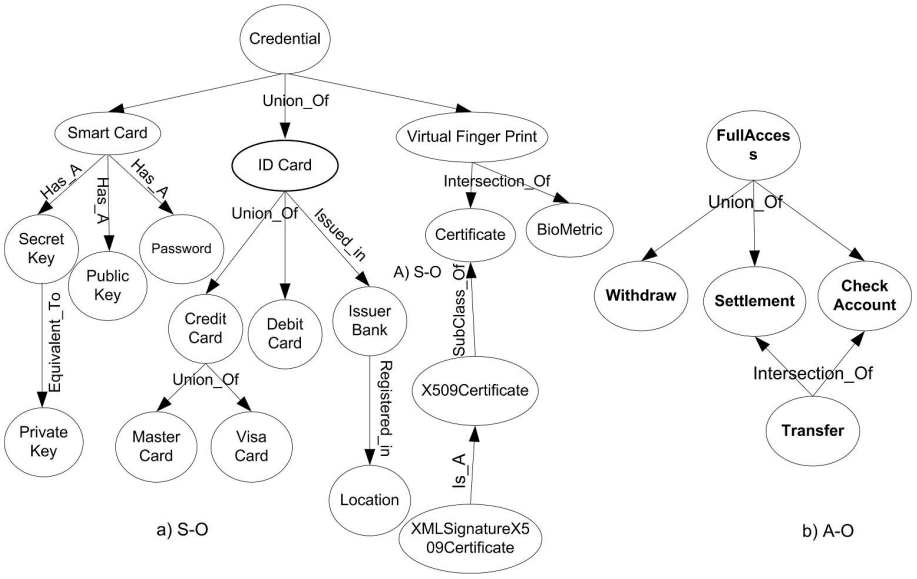

Fig. 2. a) A Credential ontology for modeling the subject domain. b) A part of executable actions on the Bank-Services ontology.

4 Semantic Authorization Inference

Different semantic relations in an ontology result in semantic authorization flow among entities in different levels of that ontology. OWL is the standard language which is proposed by W3C for representing ontologies in a machine-processable format. To automate the inference process in SBAC, we used this language since its well–defined structure lets machines automatically process the knowledge described in it, besides it supports strong semantic relations among concepts. Based on OWL, we have identified three levels: concept–level, individual–level and property–level where the semantic authorization flow can occur in each level or between different levels. To simplify the effect of semantic authorization flow in decision making, first we classify the possible semantic inferences that can occur, and then we explain different types of inferences in each category. This classification is done based on the fundamentals of OWL structure [6] which are OWL Class Axioms, Individual Axioms, Property Characteristics and Property Restriction.

– **Concept-Concept (C-C)**: Inference can be done in the level of concepts (between two concepts) in ontology. Concept constructors in OWL result in new concepts with an intrinsic semantic authorization flow. For example, when the concept 'Credit Card' is defined as the union of 'Master Card' and 'Visa Card', then access rights such as eligibility of the owner of a credit card for checking an account will be propagated to on both the owner of a 'Master Card' and the owner of a 'Visa Card'.

- **Concept-Individual (C-I)**: The semantic authorization flow from the concept level to the individual level is inevitable since all the individuals are influenced by the access conditions enforced on the concept they belong to.
- **Individual-Individual (I-I)**: Individual axioms cause this kind of authorization flow. For example the 'same as' axiom states that two individuals are semantically equal, hence the access conditions on each of them should be applied on the other one too.
- **Property-Concept (P-C)**: The semantic authorization flow from properties to concepts happens when an access right on a property is granted. A property is interpreted by a set of ordered pairs of individuals where the first individual is in the domain of the property and the latter is in the range of it. Any access right on a property can result in the same access right on the domain and range of the property. For example, when a subject can modify a property, s/he should be able to access the domain and range of the property.
- **Property-Property (P-P)**: Semantic relations between various properties can result in new properties necessary to decision making but are not explicitly mentioned in the ontology. For example, when a bank authority wants to prevent master cards supported by Asian banks from settling money in special accounts by defining $(AsianMasterCards, URI_x, -settelement)$, by having knowledge on two properties 'Issued_in' and 'Registered_in', the new property of 'Supported_by' can be made. The related SWRL rule is as follows:

$$Registered\_in(Bank_x, Asia) \wedge Issued\_in(MasterCard, Bank_x) \rightarrow Supported\_by(MasterCard, AsianBank)$$

- **Property-Individual (P-I)**: The semantic authorization flow from a property to its individuals is inevitable since all the individuals are influenced by the access conditions enforced on the property that they belong to. Moreover, property characteristics like being transitive or symmetric imply membership of some new individuals to the same property which are also affected by the access conditions defined on the property. For example, if we define the 'Support_Of' property as a symmetric property then by having the knowledge that $(Account_x, Account_y)$ is an individual of a property then it can be inferred that $(Account_y, Account_x)$ is also an individual of that property. An SWRL rule like the following can be added for the inference:

$$Support\_of(Account_x, Account_y) \rightarrow Supported\_of(Account_y, Account_x)$$

- **Concept-Property (C-P)**: When an access right on a concept is granted, then there is a semantic authorization flow from the concept to the restricted concept that is the result of property restrictions. For example, when a subject is eligible to 'Check_Balance' of some credit cards then s/he should be authorized to 'Check_Balance' of any restricted concept like $Issued\_In.Bank_x$ which returns credit cards issued in the $Bank_x$.

It is worth noting that the ontology languages in the fourth layer of the semantic web stack are not expressive enough to support all of the inference classifications that should be performed in the machine level. Using SWRL rules provides a better expressivity. But it must be taken into account that defining SWRL rules for gaining more expressiveness can not be done automatically in the machine-level; hence these types of reasoning require a human that explicitly define such rules.

4.1 Reduction to Subsumption

Different kinds of semantic relations and inference problems based on them motivated us to reduce the possible inferences on the semantic relationships in OWL DL to the general problem of *Subsumption*. Checking the subsumption property is the basic reasoning method of description logics [14]. Given two concepts C and D and a knowledge base Σ, the following illustrates that D subsumes C in Σ: $\Sigma \models C \sqsubseteq D$. This reasoning based on subsumption proves that D (the subsumer) is more general than C (the subsumee). In other terms, the concept C is considered more specific than the concept D. In SBAC, we use a variant of the subsumption relation which is represented by $\preceq$ and not only handles concepts but also handles individuals. It is defined as follows: When there is

$$A \preceq B \ = \begin{cases} A \sqsubseteq B, & if\ A\ and\ B\ are\ concepts \\ A\ Is\_A\ B, & if\ A\ is\ an\ individual\ and\ B\ is\ a\ concept \\ A\ sameAs\ B, & if\ A\ and\ B\ are\ individuals \end{cases}$$

$A \preceq B$ relation between A and B, the authorization rules enforced on B should also be enforced on A. Table 1 shows the reduction based on OWL class axioms. Table 2 is for individual axiom and Table 3 shows the reduction for OWL Property Restrictions. Table 4 shows SWRL rule definition for OWL Property Characteristics.

5 Formal Definitions of Concepts in SBAC

In this section, we present a formal definition of the topics described informally in preceding sections. SBAC is defined by the triple $(OB, AB, Oprs)$. OB stands for Ontology Base which contains decision making ontologies (OO, SO, AO). AB stands for Authorization Base that includes explicit authorization rules. $Oprs$ are the operations that can be performed on the Authorization Base.

$SBAC = (OB, AB, Oprs)$
$OB = \{Ont \mid Ont = SO \vee Ont = OO \vee Ont = AO\}$
$Ont = (C, T, \leq_C, \leq_T, R, A, \sigma_A, \sigma_R, \leq_A, \leq_R)$

Table 1. Reduction in the Scope of OWL Class Axioms

OWL Constructors	**Affected Group**	**Reduction to Subsumption**
C subClassOf D	C-C, C-I	$C \preceq D$
C equivalentClass D	C-C, C-I	$C \preceq D \wedge D \preceq C$
C disjointWith D	C-C, C-I	$C \preceq \neg D \wedge D \preceq \neg C$
C intersectionOf $C_1, \ldots, C_n$	C-C, C-I	$C \preceq C_1 \wedge \ldots \wedge C \preceq C_n$
C unionOf $C_1, \ldots, C_n$	C-C, C-I	$C_1 \preceq C \wedge \ldots \wedge C_n \preceq C$
C complementOf D	C-C, C-I	$C \preceq \neg D \wedge \neg D \preceq C$
C one of Enumeration E $\{\ldots\}$	C-C, C-I	$C \preceq E$
P1 subPropertyOf P2	C-C, C-I	$Domain(P1) \preceq Domain(P2)$ $Range(P1) \preceq Range(P2)$
P1 equivalentProperty P2	C-C, C-I	$Domain(P1) \preceq Domain(P2)$ $Range(P1) \preceq Range(P2)$ $Domain(P2) \preceq Domain(P1)$ $Range(P2) \preceq Range(P1)$

Table 2. Reduction in the Scope of OWL Individual Axioms

OWL Individual Axioms	**Affected Group**	**Reduction to Subsumption**
I1 differenetFrom I2	No Affect	–
allDifferent	No Affect	–
sameAs(I1,I2)	I-I	$I1 \preceq I2$ $I2 \preceq I1$

$AB = \{(s, o, \pm a) \mid s \in SO \ \wedge \ o \in OO \ \wedge \ a \in AO\}$
$Oprs = (CA, Grant, Revoke)$

In the definition of ontology (Ont), which is from [15], C is a set of concepts, $\leq_C$ is the subsumption relation between concepts. The other semantic relations are presented by $\sigma_R : R \rightarrow C \times C$. $\leq_R$ shows the hierarchy among Object Properties, meaning one property is subproperty of another property. T is a set of datatypes with a hierarchy of datatypes, $\leq_T$. DataType Properties are presented by $\sigma_A : A \rightarrow C \times T$ [6].

Access rights are stored in AB in the form of Authorization rules where:

$$AB \subseteq S \times O \times A$$

Definition (Authorization Rule)
An authorization rule is a triple like $(s, o, \pm a)$ where $s \in SO$, $o \in OO$, and $a \in AO$.

The knowledge base is consist of explicit authorization rules and is formally defined $AB \subseteq S \times O \times A$. An authorization rule is a triple $(s, o, +a)$ where $s \in SO$, $o \in OO$, $a \in AO$.

Definition (Operations)

Table 3. Reduction in the Scope of OWL Property Restriction

OWL Property Restriction	Affected Categories	Reduction to Subsumption
C allValuesFrom(P,D)	P–C, C–C, C–I	$C \preceq Domain(P)$ $D \preceq Range(P)$
C someValuesFrom(P,D)	P–C, C–C, C–I	$C \preceq Domain(P)$ $D \preceq Range(P)$
C minCardinality(P)	P–C, C–C, C–I	$C \preceq Domain(P)$
C maxCardinality(P)	P–C, C–C, C–I	$C \preceq Domain(P)$

Table 4. SWRL Rule Definition in the Scope of OWL Property Characteristics

OWL Property Characteristics	Has Effect	Affected Categories	SWRL Rules
TransitiveProperty	Yes	P–I, P–P	$P(a,b) \wedge P(b,c) \rightarrow P(a,c)$
SymmetricProperty	Yes	P–I, P–P	$P(a,b) \rightarrow P(b,a)$
FunctionalProperty	No	No Affect	$P(a,b) \wedge P(b,c) \rightarrow P(a,c)$
InverseOfProperty	Yes	P–I, P–P	$P(a,b) \rightarrow P^{-1}(b,a)$
InverseFunctionalProperty	No	No Affect	–

The operations are executed on AB and are for making decision about a request, granting an access right or revoking an access right and the formal definition is $Opr = (CA, Grant, Revoke)$.

- $CA(s,o,a)$: the function of decision making is $CA : S \times O \times A \rightarrow \{true, false\}$. $CA(s,o,a) = true$, if $(s,o,+a) \in AB$ or there is an authorization rule $(s_i,o_j,a_k) \in AB$ such that $(s_i,o_j,+a_k) \rightarrow (s,o,+a)$. $CA(s,o,a) = false$, if $(s,o,-a) \in AB$ or there is an authorization rule $(s_i,o_j,a_k) \in AB$ such that $(s_i,o_j,-a_k) \rightarrow (s,o,-a)$. Otherwise, due to the close policy the function returns 'False'. The reasoning '$\rightarrow$' from (s,o,a) to (s_i,o_j,a_k) can be performed on domains subject SO, object OO or action AO. Definition of function CA is as follows:

$$CA(s,o,a) = \begin{cases} True, & (s,o,+a) \in AB \vee (\exists (s_i,o_j,+a_k) \in AB : \\ & \quad (s_i,o_j,+a_k) \rightarrow (s,o,+a)) \\ False, otherwise \end{cases}$$

  Conflicts are possible in $CA(s,o,a)$ in the time of decision making. Exception rules are one of the sources of conflicts. Since for making a decision about a request two conflicting inferences can lead to different results, conflict resolution is necessary in SBAC. Inference from exception rules should have higher priority than inference from other explicit rules. Hence for resolving the conflict, the inference from the most specific rule which is the most specific exception takes precedence than other inferences. This conflict resolution policy is possible since the conflicting sources of inference are on the same inference path and comparing the conflicting rules is possible. In the cases that the conflicting rules are not comparable or in other words

they are not on the same inference path, the negative take precedence policy which gives the priority to the negative authorization rule is used for resolving the conflict.

– $Grant(s, o, a)$: Granting an authorization (s, o, a) means inserting the rule in AB . This operation is executed by the operation $Grant(s, o, a)$, which returns the Boolean value True if the rule is added and False if the rule can not be added to AB.

```
Grant(s,o,a):
    if (s,o,a) ∈ AB or CA(s,o,a) = true then
            return false
    else
            add (s,o,a)
            return True
```

– $Revoke(s, o, a)$: Revoking an authorization (s, o, a) means deleting it from AB. This operation is executed by the operation $Revoke(s, o, a)$, which returns the Boolean value True if the rule is deleted and False if the rule can not be deleted from AB.

```
Revoke(s,o,a):
    if (s,o,a) ∈ AB then
            delete (s,o,a)
            return True
    else
            return false
```

5.1 Authorization Propagation

In this section, we explain how reducing the inference problem to the subsumption problem can result in an effective way for authorization propagation in three domains of access control. In the domain of subject and object, the authorizations are propagated from subsumee to subsumer; but the propagation of access rights in the domain of actions is different from the propagation in SO and OO domains. The negative access rights will be propagated from subsumer to subsumee. It mean that subsumee can not have a positive right while the subsumer does not have it. But the positive access rights are propagated in the opposite direction. In other words, if the subsumee has a positive access right, the subsumer should also have it. The following is a formal description of the propagation mechansim:

– **Propagation in subject domain**: Given $(s_i, o, \pm a)$, If $s_j \preceq s_i$ then the new authorization rule $(s_j, o, \pm a)$ can be derived by inference from s_i to s_j, we denote this rule as $(s_i, o, \pm a) \rightarrow (s_j, o, \pm a)$.
– **Propagation in object domain**:Given $(s, o_i, \pm a)$, If $o_j \preceq o_i$ then the new authorization rule $(s, o_j, \pm a)$ can be derived by inference from o_i to o_j, we denote this rule as $(s, o_i, \pm a) \rightarrow (s, o_j, \pm a)$.

– **Propagation in action domain**:
  - Given $(s, o, +a_i)$, If $a_j \preceq a_i$ then the new authorization rule $(s, o, +a_i)$ can be derived by inference from a_i to a_j, we denote this rule as $(s, o, +a_i) \rightarrow (s, o, +a_j)$.
  - Given $(s, o, -a_j)$, If $a_j \preceq a_i$ then the new authorization rule $(s, o, -a_i)$ can be derived by inference from a_j to a_i, we denote this rule as $(s, o, -a_j) \rightarrow (s, o, -a_i)$.

6 Conclusions and Future Work

In this paper, we presented SBAC as an access control model for protecting Semantic Web resources. SBAC takes into account semantic interrelations among entities in the domains of decision making of access control. Automated decision making in SBAC for granting or denying an access request is done through inference processes based on the semantic relation among entities.

To enhance the expressiveness of the model for describing the authorization rules, more expressive logics in logic layer of Semantic Web stack can be applied. Since more expressive logics are less decidable, approaches like client based access control approaches [17] seems suitable for delegating some access control phases to the client side.

References

1. Hengartner, U., Steenkiste, P.: Exploiting information relationships for access control. In: proceeding of third IEEE International Conference on Pervasive Computing and Communications, Percom 2005, Kauai, Island HI (2005) 278–296
2. Bonatti, P.A., Duma, C., Fuchs, N., Nejdi, W., Olmedila, D., Peer, J., Shahmehri, N.: Semantic web policies – a discussion of requirements and research issues. In: ESWC 2006. (2006) 712–724
3. Samarati, P., di Vimecati, S.C.: Access control: Policies, models, architectures. In: FOSAD 2000. Volume 2171 of LNCS., Springer-Verlag (2001) 137–196
4. Qin, L., Atluri, V.: Concept-level access control for the semantic web. In: ACM Workshop on XML Security, Fairfax, VA, USA (2003) 94–103
5. Yague, M., Mana, A., Lopez, J.: Applying the semantic web layers to access control. In: Proceeding of 14th IEEE International Workshop on Database and Expert Systems Applications. (2003) 622–626
6. Patel-Schneider, P., Hayes, P., Horrocks, I.: OWL: Web Ontology Language Semantics and Abstract Syntax, W3C Recommendation (2004)
7. Hayes, P., Horrocks, I., Patel-Schneider, P., Boley, Tabet, S., Grosof, B., Dean, M.: SWRL: A Semantic Web Rule Language Combining OWL and RuleML (2004)
8. Denker, G., Kagal, L., Finin, T., Paolucci, M., Sycara, K.: Security for daml web services: Annotation and matchmaking. In: Proceedings of the 2nd International Semantic Web Conference, Sanibel Island, Florida, USA (2003)
9. Rabitti, F., Bertino, E., Kim, W., Woelk, D.: A model of authorization for next-generation database systems. ACM TODS **16**(1) (1991)
10. Prud'hommeaux, E.: W3C ACL System (2001)
11. Moses, T.: (eXtensible Access Control Markup Language (XACML), version 2.0)

12. Joshi, J.: Access-control language for multi domain environments. IEEE Internet Computing **8**(6) (2004) 40–50
13. Kagal, L., Finin, T., Joshi, A.: A policy language for a pervasive computing environment. In: Proceeding of 4th IEEE International Workshop on Policies for Distributed Systems and Networks. (2003) 63–74
14. Horrocks, I.: The fact system. In: Automated Reasoning with Analytic Tableaux and RelatedMethods: International Conference Tableaux'98, Springer-Verlag (1998) 307–312
15. Ehrig, M., Haase, P., Stojanovic, N., Hefke, M.: Similarity for ontologies - a comprehensive framework. In: Workshop Enterprise Modelling and Ontology: Ingredients for Interoperability, PAKM 2004. (2004)
16. Parsia, B., Sirin, E.: Pellet: An OWL DL Reasoner. In Moller, R., Haaslev, V., eds.: Proceedings of the International Workshop on Description Logics (DL2004). (2004)
17. Bauer, L., Schneider, M., Felten, E.: A general and flexible access-control system for the web. In: Proceedings of the 11th USENIX Security Symposium. (2002)