A Survey on Context Security Policies in the Cloud

Yiannis Verginadis and Gregoris Mentzas
Institute of Communications and Computer Systems
National Technical University of Athens
Athens, Greece

Simeon Veloudis and Iraklis Paraskakis
South East European Research Centre (SEERC)
International Faculty of the University of Sheffield,
CITY College
Thessaloniki, Greece

***Abstract*—With the pervasion of cloud computing new security risks are created. A promising approach to alleviating these risks is to provide a security-by-design framework that will assist cloud application developers in defining appropriate context-driven access control policies. This paper surveys different approaches to context-driven access control, as well as different modelling formalisms for representing access control policies. The aim of this survey is to assess the appropriateness of existing approaches for the construction of a generic security-by-design framework, in particular one which is exposed as a PaaS offering.**

Keywords-context; access control; security policy; cloud security; policy modelling

I. INTRODUCTION

Cloud computing introduces an economy-based paradigm whereby infrastructure, platform, and application resources are abstracted as services and delivered remotely, over the Internet, by a multitude of providers [1]. Its increasing adoption brings about significant benefits for enterprises and users in terms of cost savings, increased flexibility and business agility. At the same time, however, it creates new security vulnerabilities stemming mainly from the fact that corporate data reside in externally-controlled servers. Exploiting these vulnerabilities may result in data confidentiality and integrity breaches [2].

A promising approach to alleviating the security risks associated with cloud computing is to assist application developers in defining effective security controls for the sensitive data of their cloud applications [3]. To this end, we envisage a generic security-by-design framework which is provided as a PaaS solution and which guides developers through the process of defining appropriate access control policies for safeguarding their sensitive data. Such a framework bears two seminal characteristics. Firstly, it hinges upon an adequate access control scheme, one which takes into account the inherently dynamic and heterogeneous nature of cloud environments. Secondly, it captures the knowledge that lurks behind such a scheme (e.g. actions, actors, locations, environmental attributes, etc.) using a generic and extensible formalism, one which can be tailored to the particular needs of different cloud applications. This knowledge may then be used for reasoning generically about the correctness and consistency of the access control policies defined by application developers.

The first characteristic calls for the incorporation of the notion of *context* in access control policies, i.e. the consideration of dynamically-changing contextual attributes that characterise data accesses. In fact, the use of contextual information enables data owners, or administrators, to apply access control policies without any prior knowledge of the specific entities that might request to access sensitive data by considering only the circumstances under which this access should be allowed. The second characteristic calls for the adoption of a declarative approach to modelling policy-related knowledge, one which is orthogonal to the code of any particular cloud application and which can be easily adapted to suit the needs of any such application. In this respect, this paper sets out to present an overview of different approaches to context-driven access control and to policy modelling that have been reported in the literature. In particular, with respect to context-driven access control it presents work on context modelling, context detection and context awareness, highlighting their advantages and disadvantages. With respect to policy modelling, a state-of-the-art analysis is performed focusing on the benefits and drawbacks of various efforts revolving around the declarative representation of policy-related information. We believe that such a literature survey will shed light on the appropriateness of existing approaches for the construction of a generic security-by-design framework such as the one discussed above.

The rest of this paper is organized as follows. Section II presents work on context-driven access control that are related to different access control models such as role-based access control, mandatory access control, discretionary access control and attribute-based access control. Section III presents different strands of declarative policy description formalisms mainly focusing on syntactic policy descriptions and semantically-rich policy modelling. Finally, Section IV presents conclusions and future work.

II. SECURITY-RELATED CONTEXT

In this section, we describe research work relevant to context modelling and we classify existing security and privacy related shortcomings.

A. *Context-driven Access Control*

Among the most significant security related concerns in dynamic and heterogeneous environments especially in cloud-enabled systems is the access control that should be able to consider most of the dynamic aspects of such environments. The emerging and ubiquitous computing environments need security control that is easily adaptable to the changing user or environmental contexts. Context information used in an access control decision can be defined as any relevant information about the state of a relevant contextual entity or the state of any relevant relationship between different relevant entities at a particular time that should be taken into account before granting or rejecting a specific access request. From this perspective, context-awareness relates to the use of this context information for access control decision making. In the literature, there are three basic access control models [4], namely Discretionary Access Control (DAC), Mandatory Access Control (MAC), and Role-Based Access Control (RBAC). All these models are known as identity based access control models where user (subjects) and resources (objects) are identified by unique names [5]. In the literature, a fourth type has been recognised, the Attribute Based Access Control (ABAC) that is discussed below. It is true that most of the traditional security services are context insensitive; i.e. they do not adapt their security access control to a changing context. In this section, we discuss the most interesting research approaches that use context as a first class entity for access control decision making without restricting them to only the cloud-based ones.

B. RBAC Related Approaches

The basis of RBAC is the concept of a role which constitutes a grouping mechanism for categorizing individual users (called subjects) based on various properties (e.g. job title, user functions, responsibilities etc.). Each subject has a role set, which consists of all the roles that the subject has been authorized to use. Although RBAC is very useful for modelling access control in a variety of applications, its roles are inherently too static in the sense that they cannot be used at run-time to capture security-relevant context from the environment and ultimately have a dynamic impact on access decisions. RBAC thus lacks, in general, support for expressing access control conditions that refer to the state of a system, e.g. the state of a protected resource, parameter values, date or time [6]. In order to alleviate the disadvantages of such static models, a number of solutions have been proposed for context-based access control. One of them is the Organization Role Base Access Control (ORBAC) model. ORBAC is an access control model in which authorization is given to users depending on their role in an organization in a given context [7]. Another more interesting approach from the context perspective is the Generalized RBAC (GRBAC) [8]. The GRBAC paradigm incorporates the concept of environment roles. Environment roles capture environmental information, such as time of day or weather conditions, which can be used to mediate access control. Convington et al. [9] were among the first to apply the GRBAC paradigm in applications for intelligent homes ("the Aware Home"). Furthermore, GRBAC models, support richer contextual information for impacting the result of an access request than just considering only time and location aspects that other approaches support. The main drawback of this work is that it proposes a domain specific environment role hierarchy that it is not easily extensible and manageable in heterogeneous domains while it doesn't support the fine-grained modelling of different data objects. In general, although GRBAC offers higher expressiveness making it theoretically more suitable for context-aware authorization schemes, in practice it becomes cumbersome to maintain as there are potentially large amounts of environment roles to manage.

Other existing models also known as Context-aware access control (CAAC) are predominantly based on RBAC [10], having some of them typically targeting a specific domain. For example, Zhang and Parashar [10] have proposed a Dynamic RBAC model that extends the role based access control model and 'dynamically' adjusts static Role Assignments and Permission Assignments based on context information. Such models, however, have not been designed to provide fine-grained data access control, e.g. by providing the ability to specify different access rules for different rows of a database. In addition, other approaches like [11] incorporate only specific types of contexts such as location and time. Kulkarni et al. [12] have proposed a Context-aware RBAC (CA-RBAC) model for pervasive applications that consider user and resource attributes as context constraints. He et al., [13] considered access control for Web services based on the roles and introduced a CAAC policy model considering the user, resource and environment concepts. Toninelli et al. [14], proposed a CAAC approach, which provides resource access permission on the basis of resource availability, roles of user, location and time. It involves an ontology-based framework that includes both context and policy models. The disadvantage of the above mentioned approaches is that they only consider specific types of contexts, which are not sufficient and generic enough to be used in dynamic cloud environments. Lodderstedt et al. [6], presented an RBAC based approach in which they proposed SecureUML, a modelling (annotation) language designed to integrate information relevant to access control into application models defined in UML. To cover inefficiencies of traditional RBAC models, they also introduced the concept of authorization constraints. These are defined in the Object Constraint Language (OCL) and express preconditions for granting access to one or more operation on particular resources. Although these preconditions take into account the dynamic state of the resource, the current call, or the environment, they are not based on an extensible and reusable context model, rendering this approach rather static for the requirements of cloud-based systems.

In a similar vain, the work in [15] proposed an ontology-based context model, named Context Ontology for Access Control (COAC), for representing and capturing different types of context information in a systematic way. The added value of this very interesting work is the fact that this model allows for reasoning about high-level implicit context information that is not directly available but can be derived from other information. In order to express their context model, the OWL language is used and is extended with SWRL for inferring implicit context with user-defined rules. The specific ontology involves two kinds of context entities; the core (i.e. user, resource, owner, role) and environmental (i.e. RelationshipInfo, StatusInfo, ProfileInfo, LocationInfo, TemporalInfo, and HistoricalInfo) that are relevant to the access request. Based on that model the authors also introduced a context-aware policy ontology called CAPO (expressed also using OWL and SWRL) for defining and enforcing access control policies and taking into account relevant contextual information. This policy model provides context-aware access control decisions and has been demonstrated with a prototype in the healthcare domain.

Nevertheless, ontology-based reasoning in OWL is not considered efficient in inferring the high-level implicit contexts especially in highly dynamic and heterogeneous environments like cloud-based systems.

C. MAC and DAC Related Approaches

MAC and DAC are among the first access control models that have been used in highly security sensitive production environments. Solutions that are based on MAC access control are inherently inappropriate as a basis for CAAC as they involve rigid and static methods that are used for highly-secure military-type of settings. An effort that uses such underlying security model is that of Jürjens [16] that proposed a specification of requirements on confidentiality and integrity in analysis models based on UML.

DAC access control models define access control matrices whose rows and columns correspond to subjects and objects respectively while their intersection points corresponds to a set of allowed access operations. Such sets of allowable operations are often implemented in terms of Access control lists (ACL) where each object is associated with an ACL that determines both positive permissions and negative permissions. Negative permissions enjoy higher precedence than positive ones. In addition to that, DAC models are commonly used with groups, whereby users with similar access rights are formed into groups to which there are assigned specific permissions. Group policies are typically expressed in terms of Group Policy Objects (GPOs), i.e. sets of configurations that determine various attributes pertaining to user accesses such as maximum number of allowable failed logins, password strength, per-object access rights etc. Typically, access control permissions are defined statically. A related approach has been proposed by Bertino et al. [17], who investigated support for temporal authorizations in database systems. They have examined both periodic and non-periodic authorizations. Their authorization language is defined in a low level format, which inherently limits its usefulness and reusability capabilities.

D. ABAC Related Approaches

The above mentioned access control models, along with their extensions, can be considered as special cases of Attribute Based Access Control (ABAC). According to NIST [18] ABAC is a logical access control methodology where authorization to perform a set of operations is determined by evaluating attributes associated with the subject, object, requested operations, and, in some cases, environment conditions against policy, rules, or relationships that describe the allowable operations for a given set of attributes. The key difference of ABAC is the fact that the concept of provided policies can express a complex Boolean rule set that can evaluate many different attributes. One example of an access control framework that is consistent with ABAC is the OASIS eXtensible Access Control Markup Language (XACML). It has been used for describing declarative access control policies encouraging the separation of the access decision from the point of use. Bhatti et al. [19], described a framework for enforcing role based access control in dynamic XML-based web services. Their solution includes concepts of roles and context.

A number of other research efforts (e.g. [20, 21] have further extended the Attribute-based Access Control (ABAC) approach to provide access control to software services in a context-aware manner. Such an example is the concept of Location-aware access control (LAAC), which allows a system to grant, or deny, access to users based on their physical location. LAAC models typically extend the three basic access control models [4]. Even though LAAC protocols have been studied extensively [22], there is a clear lack of schemes that determine user access not only on the basis of the users' physical location and provided credentials, but also according to additional pertinent contextual information. Corradi et al. [20], proposed a CAAC model for ubiquitous environments, where permissions are directly associated with contexts (e.g. user location, user activities, user device, time, resource availability and resource status). Hulsebosch et al. [21], proposed a Context-sensitive access control (CSAC) framework and suggested a process whereby the service provider can define various security rules based only on the context of the user and not her actual identity. They actually define access control on the basis of an inextricable relationship between user/device and service, and they proposed verification methods for anonymous access based on location and service usage history patterns. In particular, the subject authentication is based on verifying whether or not the situational context claimed is a valid context attribute of the subject. Nevertheless, the security of the proposed model is questionable since the authors fail to provide a security analysis of their proposed model. Apart from that, authors do not describe the adversarial model under which their protocol is considered secure, thus it is not clear if the use of the proposed technique in real life applications will lead to security breaches. Moreover, a generic limitation that these approaches present is that they consider only a limited set of contexts which limits the functionality that can be offered to the end-user.

Chen et al. [23], defined Context Broker Architecture (CoBrA). CoBrA is a context ontology based on OWL which only covers domain specific context elements in a smart space environment, while it has no explicit support for modelling general contexts in heterogeneous environments. CoBra, focuses on modelling static physical space with relative stable lower context data resources. In CoBrA the authors used the Standard Ontology for Ubiquitous and Pervasive Applications (SOUPA) [24]. The SOUPA ontology is expressed using OWL and includes modular component vocabularies to represent intelligent agents, in pervasive context-aware systems, with associated beliefs, desire, and intentions, time, space, events, user profiles, actions, and policies for security and privacy. In [25] Onto-ACM is proposed, a semantic analysis model that can address the difference in the permitted access control between service providers and users. This model was presented as an intelligent context-aware access scheme for proactively determining the level of resource access based on ontology reasoning and a semantic analysis method which distinguishes between user and administrator policies. In addition to that, a context ontology was discussed for user authentication and authorization consisting of identity, physical, preference, and other information. Wang et al., [26] proposed an OWL encoded CONtext ONtology (CONON) for modelling context information in pervasive environments. CONON has been designed having particular applications in mind (smart homes) and is not considered flexible for offering fine-grained access control in other application domains. Henricksen et al. [27] developed a Context Modeling Language (CML) and a tool that translates CML-based context models to an OWL representation for the purpose of utilizing the OWL technology. However, such existing general context models do not provide direct support for concepts related to access control. A recent CAAC framework for the Web of data is grounded on two ontologies which deal with the core access control policy concepts and the context concepts [28]. Despite the fact that it supports access control and considers three important dimensions of context (user, device and environment) the framework does not have the capability of inferring high-level implicit contexts. In general, ABAC and their extensions avoid the need for capabilities (operation/object pairs) to be directly assigned to subject requesters or to their roles or groups before the request is made.

III. Related Work on Policy Modeling

The purpose of policies is to generally determine and guide the manner in which entities within a particular domain act. To this end, they provide a set of unambiguous rules which are interpreted by enforcement mechanisms and which constrain the behaviour of the entities. By separating policies from the implementation of the enforcement mechanisms, it is possible to provide different behavioural constraints for the entities without having to change the implementation of the enforcement mechanisms. Such a separation, however, requires a declarative representation of policies, one which is orthogonal to the code of the enforcement mechanisms. This section sets out to provide an overview of existing formalisms for the declarative representation of policies. More specifically, it outlines approaches that: focus on syntactic descriptions; propose semantically-rich representations. Our overview is not limited to security policies but considers policies in general.

A. *Syntactic Policy Description*

Syntactic descriptions were introduced along with the Service Oriented Architecture (SOA) model as part of a standardisation effort aiming, primarily, at facilitating interoperable data exchanges in interactions. In the realm of policies and policy-based applications, syntactic descriptions promote a declarative approach to policy expression, one which aims at replacing a trend whereby policies are encoded imperatively, as part of the same software that checks for their compliance.

Several markup languages have been proposed for the declarative description of policies, some prominent examples being RuleML [29], XACML [30], SAML [31] and WS-Trust [32]. These provide XML-based syntaxes for expressing policy rules. A reference monitor, or policy decision/enforcement point, is then employed to parse these rules and determine whether a particular actor is allowed to perform certain actions. Nevertheless, such syntactic descriptions fail to capture the knowledge lurking behind policies. In this respect, they are merely data models that lack any form of semantic agreement beyond the boundaries of the organisations that developed them. Any interoperability relies on the use of vocabularies that are shared among all parties involved in an interaction. This has a number of limitations: (i) it leads to ad-hoc reasoning about policy compliance, one which is tied to the specific vocabularies that express the rules according to which the reasoning takes place; (ii) it limits the reusability and portability of policies; (iii) it precludes the identification of inter-policy relations; (iv) it limits the ability to perform policy governance.

B. *Semantically-rich Policy Description*

In order to overcome the aforementioned limitations, semantically-rich approaches to the specification of policies have been brought to the attention of the research community [33-36]. These generally embrace Semantic Web representations for capturing what we term action-oriented policies, i.e. policies which control when a particular actor can perform a specified action on, or through the use of, a certain resource. More specifically, these approaches employ ontologies in order to assign meaning to actors, actions and resources. Being "a formal, explicit specification of a shared conceptualization" [37], an ontology provides a flexible, formal, and unambiguous means of agreement upon the semantics of concepts, and their interrelations, in a given domain of discourse.

It becomes evident that, in contrast to syntactic policy descriptions which aim at devising purpose-built vocabularies for expressing the rules that implement policies, semantically-rich descriptions introduce an extra layer of abstraction which captures the knowledge that dwells in policies. The rules that implement these policies can then be generated from this knowledge in an automated manner, and can be expressed using any suitable syntactic description language. This brings about the following advantages: (i) it paves the way for the construction of policy-enforcement mechanisms able to reason about policy compliance generically and orthogonally to any particular syntactic representation of policy rules; (ii) it enables the establishment of associations between operational-level policies to other policies at the same or higher (strategic) level of abstraction, thus enabling the identification of inter-policy relations such as inconsistent policies, and overlapping policies. Such relations are important as they may lead to erroneous decisions (e.g. in the case of conflicting policies), or may degrade system performance (e.g. in the case of overlapping policies); (iii) it promotes the portability, visibility, and reusability of policies; (iv) it facilitates policy updates and maintenance and hence policy governance.

In [33], the authors presented KAoS – a general-purpose policy management framework which exhibits a three-layered architecture comprising: (i) a human interface layer, which provides a graphical interface for policy specification in natural language; (ii) a policy management layer, which uses OWL to encode and manage policy-related knowledge; iii) a policy monitoring and enforcement layer, which automatically grounds OWL policies to a programmatic format suitable for policy-based monitoring and policy enforcement.

In [34] the authors proposed Rei – a policy specification language expressed in OWL-Lite [38]. It allows the declarative representation of a wide range of policies which control which actions can be performed, and which actions should be performed, by a specific entity. Furthermore, it defines a set of concepts (rights, prohibitions, obligations, and dispenations) for specifying and reasoning about access control rules. In this respect, it provides an abstraction which allows the specification of a desirable set of behaviours which are potentially understandable – hence enforceable – by a wide range of autonomous entities in open and dynamic environments.

In [35], POLICYTAB is proposed for supporting trust negotiation in Semantic Web environments. POLICYTAB advocates an ontology-based approach for describing policies that drive a trust negotiation process aiming at providing controlled access to Web resources. Such policies essentially specify the credentials that a party has to present for performing an action on a resource owned by another party. A plugin for the ontology editor Protégé is provided in order to facilitate policy specification.

In [36], the authors recognise that cloud computing, and in particular the concept of multi-tenancy, calls for policy-driven access control mechanisms. Nevertheless, the different types of access control policies, their complex nature, and the lack of effective policy analysis mechanisms, often lead to inconsistent and/or overlapping policy sets, and thus to error-prone access control mechanisms. In this respect, they propose an ontology-based framework to capture the common semantics and structure of different types of access control policies (e.g. XACML policies, firewall policies, etc.), and facilitate the process of detecting anomalies in these policies. Their ontology captures the underlying domain concepts involved, the policy structure, and the policy attributes. Particular types of access control policies are obtained by appropriately instantiating the ontology.

IV. Conclusions AND Future Work

We have presented a state-of-the-art analysis of different approaches to CAAC and to policy modelling. Regarding CAAC, the existing RBAC-based approaches either do not cover all relevant contextual elements with a reusable security related context model, or are proven hard to maintain in dynamic environments where users often switch roles [7]. On the other hand, the access control policies that can be implemented in ABAC are limited only by the computational language and the richness of the available attributes. Thus, they present the appropriate flexibility and dynamic access control that is highly desirable and generally missing in the heterogeneous cloud computing domain.

Regarding policy modelling, the existing semantically-enhanced approaches rely on bespoke, non-standards-based, ontologies for the representation of policies. They therefore generally lack the expressivity for addressing the business details of cloud applications and thus of the access control policies that these are amenable to. In addition, the reliance of these approaches on OWL, despite the obvious benefits stemming from the rich set of properties that OWL offers, raises concerns about the degree to which these approaches are lightweight, hence about their performance.

Taking these findings into account, in the future we plan to construct a novel ABAC-based CAAC model for capturing policy-related knowledge. The model will conceptualize all contextual attributes that must be taken under consideration for controlling access to individual data objects based on their sensitivity level. In order to avoid the use of bespoke, non-standards-based, ontologies for the representation of such knowledge, the model will be expressed in Linked USDL [39] and, in particular, in Linked USDL's Security profile. Linked USDL is a lightweight ontology which provides an RDF [40] vocabulary for the description of the business aspects of policies and services. By drawing upon widely-adopted vocabularies such as GoodRelations[1], SKOS[2], and FOAF[3], Linked USDL promotes knowledge sharing whilst increasing the interoperability, hence the generality, of our approach. In addition, by embracing Linked Data as the core means for capturing facts about people, organisations, resources, and services, Linked USDL provides a flexible, general-purpose, easily extensible ontological framework that can be tailored to suit the particular security needs of different cloud applications.

Our model will be used as part of the PaaSword framework – an envisaged security-by-design framework that will provide storage protection mechanisms for maximizing and fortifying the trust of individual, professional and corporate users to cloud services and applications [3]. More specifically, the model will be used for guiding a developer in defining a set of effective access control and organizational policies in the form of Data Access Object annotations in the persistence layer of cloud applications.

Acknowledgment

The research leading to these results has received funding from the EU's H2020 programme under grant agreement No 644814, the PaaSword project (www.paasword.eu).

References

[1] L.M.Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, “A break in the clouds: Towards a cloud definition,” SIGCOMM Comput. Commun. Rev., vol 39, no 1, pp. 50 — 55, 2008.

[2] The Notorious Nine Cloud Computing Top Threats in 2013, Cloud Security Alliance, 2013.

[3] Y. Verginadis, A. Michalas, P. Gouvas, G. Schiefer, G. Hübsch and I. Paraskakis, “PaaSword: A Holistic Data Privacy and Security by Design Framework for Cloud Services,” in *Proceedings of the 5th International Conference on Cloud Computing and Services Science (CLOSER 2015)*, Lisbon, Portugal, 2015.

[4] M. Decker, Modelling of Location-Aware Access Control Rules, IGI Global, 2011.

[1] http://www.heppnetz.de/ontologies/goodrelations/v1.html

[2] http://www.w3.org/2004/02/skos/

[3] http://www.foaf-project.org/

[5] A. Khan, “Access control in cloud computing environment,” *ARPN Journal of Engineering and Applied Sciences,* 2012.

[6] T. Lodderstedt, D. A. Basin and J. Doser, “SecureUML: A UML-Based Modeling Language for Model-Driven Security,” in *In Proceedings of the 5th International Conference on The Unified Modeling Language (UML'02),* 2002.

[7] N. Boustia and A. Mokhtari, “Representation and Reasoning on ORBAC: Description Logic with Defaults and Exceptions,” in *Proceedings of the 2008 Third International Conference on Availability, Reliability and Security (ARES'08),* 2008.

[8] M. Moyer and M. Ahamad, “Generalized Role-Based Access Control,” in *In Proceedings of the The 21st International Conference on Distributed Computing Systems (ICDCS'01),* 2001.

[9] M. J. Covington, W. Long, S. Srinivasan, A. K. Dev, M. Ahamad and G. D. Abowd, “Securing context-aware applications using environment roles,” in *Proceedings of the sixth ACM symposium on Access control models and technologies (SACMAT'01),* 2001.

[10] G. Zhang and M. Parashar, “Context-Aware Dynamic Access Control for Pervasive Applications,” in *Proceedings of the Communication Networks and Distributed Systems Modeling and Simulation Conference (CDNS'04),* San Diego, CA, USA, 2004.

[11] S. M. Chandran and J. B. D. Joshi, “LoT-RBAC: a location and time-based RBAC model,” in P*roceedings of the 6th international conference on Web Information Systems Engineering (WISE'05),* 2005

[12] D. Kulkarni and A. Tripathi, “Context-aware role-based access control in pervasive computing systems,” in *Proceedings of the 13th ACM symposium on Access control models and technologies (SACMAT'08),* Vienna, Austria, 2008.

[13] Z. He, L. Wu, H. Li, H. Lai, Z. Hong, "Semantics-based access control approachfor web service". JCP 6, 1152-1161, 2011

[14] A. Toninelli, R. Montanari, L. Kagal and O. Lassila, “A semantic context-aware access control framework for secure collaborations in pervasive computing environments,” in *Proceedings of the 5th international conference on The Semantic Web (ISWC'06),* 2006.

[15] A. S. M. Kayes, J. Han and A. Colman, “An Ontology-Based Approach to Context-Aware Access Control for Software Services,” in *Proceedings of the Web Information Systems Engineering (WISE'13),* 2013.

[16] J. Jürjens, “Towards development of secure systems using UMLsec,” in *Proceedings of the 4th International Conference on Fundamental Approaches to Software Engineering (FASE'01),* 2001.

[17] E. Bertino, C. Bettini, E. Ferrari and P. Samarati, “A temporal access control mechanism for database systems,” *IEEE Transactions on Knowledge and Data Engineering,* pp. 67-80, 1996.

[18] V. C. Hu, D. Ferraiolo, R. Kuhn, A. Schnitzer, K. Sandlin, R. Miller and K. Scarfone, “Guide to Attribute Based Access Control (ABAC) Definition and Considerations,” NIST, 2014.

[19] R. Bhatti, E. Bertino and A. Ghafoor, “A Trust-based Context-Aware Access Control Model for Web-Services,” *Journal of Distributed and Parallel Databases,* pp. 83-105, 2005.

[20] A. Corradi, R. Montanari and D. Tibaldi, “Context-based access control management in ubiquitous environments,” in *Proceedings of the Network Computing and Applications, Third IEEE International Symposium (NCA'04),* 2004.

[21] R. J. Hulsebosch, A. H. Salden, M. S. Bargh, P. W. G. Ebben and J. Reitsam, “Context sensitive access control,” in *Proceedings of the tenth ACM symposium on Access control models and technologies (SACMAT'05),* 2005.

[22] A. v. Cleeff, W. Pieters and R. Wieringa, “Benefits of Location-Based Access Control: A Literature Study,” in *roceedings of the 2010 IEEE/ACM Int'l Conference on Green Computing and Communications & Int'l Conference on Cyber, Physical and Social Computing (GREENCOM-CPSCOM '10),* 2010.

[23] H. Chen, T. Finin and A. Joshi, “Semantic web in the context broker architecture,” in *Proceedings of the Second IEEE International Conference on Pervasive Computing and Communications (PerCom'04),* 2004.

[24] H. Chen , T. Finin and A. Joshi, “The SOUPA Ontology for Pervasive Computing,” in *Ontologies for Agents: Theory and Experiences,* BirkHauser, 2005, pp. 233—258

[25] C. Choi, J. Choi and P. Kim, “Ontology-based access control model for security policy reasoning in cloud computing,” *The Journal of Supercomputing,* pp. 711-722, 2014.

[26] X.H. Wang, D.Q. Zhang, T. Gu, H.K Pung, "Ontology based context modeling and reasoning using OWL", *In Proceedings of the 2nd IEEE Annual Conference on Pervasive Computing and Communications Workshops*, pp. 18 - 22, 2004

[27] K. Henricksen, J. Indulska and A. Rakotonirainy, “Modeling Context Information in Pervasive Computing Systems,” in *In Proceedings of the First International Conference on Pervasive Computing (Pervasive'02),* 2002

[28] L. Costabello, S. Villata and F. Gandon, “Context-Aware Access Control for RDF Graph Stores,” in *In Proceedings of the 20th European Conference on Artificial Intelligence (ECAI'12),* 2012.

[29] “Specification of Deliberation RuleML 1.01,” 2015. [Online]. Available: http://wiki.ruleml.org/index.php/Specification_of_Deliberation_RuleML_1.01. [Accessed 30 September 2015].

[30] “eXtensible Access Control Markup Language (XACML) Version 3.0.,” 22 January 2013. [Online]. Available: http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html. [Accessed 30 September 2015].

[31] Security Assertions Markup Language (SAML) Version 2.0. Technical Overview,” 25 March 2008. [Online]. Available: https://www.oasis-open.org/committees/download.php/27819/sstc-saml-tech-overview-2.0-cd-02.pdf [Accessed 30 September 2015].

[32] WS-Trust 1.3,” 2007 March 2007. [Online]. Available: http://docs.oasis-open.org/ws-sx/ws-trust/200512/ws-trust-1.3-os.doc [Accessed 30 September 2015].

[33] A. Uszok, J. Bradshaw, R. Jeffers, M. Johnson, A. Tate, J. Dalton and S. Aitken, “KAoS Policy Management for Semantic Web Services,” *IEEE Intel. Sys.,* vol. 19, no. 4, pp. 32 - 41, 2004.

[34] L. Kagal, T. Finin and A. Joshi, “A Policy Language for a Pervasive Computing Environment,” in *4th IEEE Int. Workshop on Policies for Distributed Systems and Networks (POLICY '03),* 2003.

[35] W. Nejdl, D. Olmedilla, M. Winslett and C. C. Zhang, “Ontology-Based policy specification and management,” in *Gómez-Pérez, A. and Euzenat, J. (eds.) ESWC'05*, 2005.

[36] H. Hu, G.-J. Ahn and K. Kulkarni, “Ontology-based policy anomaly management for autonomic computing,” in *7th International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom),* 2011.

[37] T. Gruber, “Towards principles for the design of ontologies used for knowledge,” *International Journal Human-Computer Studies,* vol. 43, no. 5-6, pp. 907--928, 1995.

[38] “OWL Web Ontology Language Reference. W3C Recommendation,” 10 2 2004. [Online]. Available: http://www.w3.org/TR/owl-ref/ [Accessed 30 September 2015].

[39] “Linked USDL,” [Online]. Available: http://linked-usdl.org/ [Accessed 30 September 2015].

[40] “RDF 1.1 XML Syntax,” 25 February 2014. [Online]. Available: http://www.w3.org/TR/2014/REC-rdf-syntax-grammar-20140225/. [Accessed 30 September 2015.