# Deploying Trust Policies on the Semantic Web

## (Position Paper)

Brian Matthews and Theo Dimitrakos

CCLRC Rutherford Appleton Laboratory
Didcot, Oxon, OX11 0QX, UK
b.m.matthews@rl.ac.uk, t.dimitrakos@rl.ac.uk

**Abstract.** Automated services over open distributed systems have been advocated as the future for conducting business. However, in order for such services to be successfully deployed, confidence in the system and its participants needs to established. The use of trust policies and trust management has been proposed to raise the level of trust whilst still maintaining a practical level of automation. To define, deploy and enforce trust policies a level of common vocabulary and understanding between participants needs to be established. The Semantic Web initiative of the W3C has been developing common languages and tools to allow the exchange and processing of common vocabularies, in the form of a graph-based description format, and an ontology language. In this paper we discuss the steps required to establish a policy-based service architecture, discuss the role of the Semantic Web initiative in enabling the practical deployment of this architecture, and highlight the work required to enable this.

## 1    Introduction

Automated services delivered over open distributed systems are seen as the next major development in science, commerce and government. However, this service vision will only come about if enterprises and individuals are confident that it will deliver services 'as advertised', not only in terms of functionality but also in the behaviour of participating agents. For example, agents can reasonably be expected to prevent unauthorised access to information and consequent malicious action, to respect confidentiality and privacy of all parties, and to supply accurate and timely results. These requirements are particularly challenging when the services depend on multiple contributing providers, who may not be known to the consumer of the overall service. Participants may gain by sharing resources, while being concerned about protecting their assets and reputation. Confidence that a party will behave well comes from a combination of **trust** between parties, and **control measures** that constrain behaviour.

In this new world, collaborations are fluid, linked by networks of trust and contracts, mediated via brokering and monitoring agents. Contracts formalise agreements among partners, stating the rights and obligations of the parties as well as sanctions to be applied if obligations are not fulfilled or rights exceeded. Contracts may then form the basis of behavioural norms and control policies applied under it.

Dependence on trust (that a partner will abide by agreements and policies) in such collaborations increases the efficiency of collaborations, but makes the collaborators more vulnerable. In contrast, enforcement of policies and contracts limits the risk of malicious or unintentional damage, at the price of increased overheads. The level of trust changes with time and with the importance of the interaction, and prescriptive contracts and policies cannot predict all possible behaviours in dynamic open systems. Consequently, trust and contract management frameworks need to be complemented by adaptive deployment mechanisms that resolve conflicts and support renegotiation and amendment in real time.

Existing tools and techniques for security management rely heavily on human intervention from system administrators and systems security officers using separate management applications in order to effect changes to the security configurations in response to security relevant events. In establishing dynamic collaborations on-demand, the scale, impact and frequency of changes increases dramatically and the variety of security mechanisms employed by the partners further impedes their deployment. Consequently, security management must become autonomic and adaptation must occur automatically in real-time, rather than through human intervention. Furthermore, autonomic security management will have to be complemented by extensible and machine processable standards for negotiating, validating and amending collaboration agreements, encoded by means of electronic contracts, which can be autonomically enacted by the platform.

Such extensible and machine processable standards require the development of common vocabularies and negotiation protocols. The Semantic Web initiative of the World Wide Web Consortium (W3C) is an effort to provide common machine-readable data onto the web, by allowing common vocabularies and conventions to be defined to describe web accessible resources. The aim of the Semantic Web is to be: *"... an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation"*[1]. The major components of the Semantic Web are the Resource Description Framework [13] for assigning properties and values to web resources, and for defining simple vocabularies, and the Web Ontology Language OWL [16], for defining more expressive vocabularies with additional constraints. Many tools and applications have now been produced on top of these basic Semantic Web tools.

In this paper we propose that the Semantic Web of the W3C provides an underlying framework to allow the deployment of a service architecture, as first suggested in [5]. We provide a survey of existing work which has been carried out to achieve this, and suggest how this and further work may be drawn together to support a complete Semantic Web enabled policy-based service architecture.

## 2 A Service Architecture

A service based architecture which is augmented with trust policies and trust management would have the following stages.

1. **Semantic Web Services:** in addition to publishing their interfaces, Web Services would need to publish statements describing their intended or normative behaviour.

These statements should be given common, machine processable, extensible semantics that support judgment of :
- whether a service can perform a given task;
- the relative ranking of a set of services with respect to basic QoS criteria.
2. **Trust policy publication:** service providers will publish policies for their use, detailing the obligations, privileges and expected levels of service which a user should accept before using the service.
3. **Service discovery:** users of services will seek services which satisfy their requirements, via automated search and negotiation mechanisms, to find the "best available" service. The criteria of "best" should include not only the functional requirements of the service delivering the service to the required quality, but also include both whether the published policy is acceptable to the user, and the reputation of the service provider, using any previous experience of using the provider, and any recommendations or guarantees provided by third parties.
4. **Service negotiation:** once a service has been selected, there needs to be a negotiation between service and user. As part of this process, the policies of both parties have to be interrogated and a contract of use established. As part of this process, a conversation needs to take place between the parties, establishing a mutually intelligible vocabulary of terms for data and process descriptions. This negotiation may involve third parties (brokers, guarantors, service framework providers etc) which may facilitate the relationship and foster trust between the parties.
5. **Experience monitoring:** during the execution of the service, which may be over a long period, its progress is monitored. The experience of the quality of the service may modify the relationship between the parties. For example, if the experience so far is good, then the parties may relax restrictions for the remainder of the service.
6. **Policy Enforcement:** policy statements need to be interpreted into lower-level rules which are then enforced at each network end-point.
7. **Service review:** after the end of the service the parties will review the progress of the service, and modify appropriately the trust evaluation of the other party, and modify the policies it will apply in future collaborations.

Thus at all stages of the process, a vocabulary describing the service needs to be established and exchanged between the parties.


# 3 Using the Semantic Web

We consider how the Semantic Web can support the deployment of the policy-augmented service architecture described above.


## 3.2 Policy publication and enforcement

Web Services standards for SOAP-based message security and XML-based languages for access control (e.g. XACML [19]) are emerging. The use of XML as a basis for expression specification has the advantage of extensibility. Its semantics however are mostly implicit as meaning depends on a shared understanding derived from human

consensus, and allow incompatible representation variations. Semantic Web-based policy representations could be mapped to lower level XML representations if required by an implementation.

Some initial efforts in the use of Semantic Web representations for basic security applications (authentication, access control, data integrity, encryption) have begun to bear fruit. For example, Denker et al. [4] have integrated a set of ontologies (credentials, security mechanisms) and security extensions for Web Service profiles with the CMU Semantic Matchmaker. Kagal et al. [9] are also developing Rei, a Semantic Web based policy language. Furthermore, KAoS services and tools allow for the specification, management, conflict resolution, and enforcement of policies within the specific contexts established by complex organizational structures represented as domains [2; 8; 17; 18]. A comparison of KAoS, Rei, and more traditional policy approaches such as Ponder can be found in [15].

The KAoS policy ontology distinguishes between authorizations and (state or event triggered, conditional) obligations. Other policy constructs, including delegation and role-based authorisations, are built out of the basic primitives of domains and the basic policy types. "Action" is defined as the ontological class used to classify instances of intended or performed actions. Applicability of action instances relates to a policy (instance) through the association of the corresponding classes. The use of OWL enables reasoning about the controlled environment, policy relations and disclosure, policy conflict detection, and harmonization, as well as about domain structure and concepts exploiting the description logic subsumption and instance classification algorithms. Taking advantage of OWL, platform/application-specific ontologies are easily loaded on top of the core policy classes.

KAoS provides a powerful tool-set that appears to be capable to address publication and deployment of complex policies for Semantic Web Services. However the incorporation of trust metrics and a distributed enforcement and performance assessment schemes remain the main challenges, in addition to the production of a critical mass of domain/application-specific ontologies to allow its uptake and validation in large scale systems. With repect to the latter there is an ongoing effort to adapt KAoS for use in Grid Computing environments in conjunction to OGSA [8].

### 3.3    Service Discovery

In order for a new service to be used it needs to be discovered and a mapping needs to be established between the requirements of the client and the capabilities of the service. On the service side, discovery is facilitate in the presence of a set of semantic descriptions. WSDL descriptions can be used to support this, but they fall short in providing any unambiguous semantic content for the service interface description they provide; OWL has been used to provide descriptions of the functionality of web services, (e.g. DAML-S [3]), but not as yet to describe their quality of service.

On the client side, the client objectives must also be given semantics in order to enable achieving a "sufficiently good" similarity between objectives of requestor and the capabilities of the service, advertised by its provider. Generally, a match can determined by heuristic algorithms, aided by domain-specific ontologies that define the terms used for service description as well as the objectives of the requestor.  Again, there is a need to extend this work to non-functional requirements.  P3P [10, 12] adds

policies and requirement of the client with respect to Privacy; this would need to be extended to express the wider quality-of-service expectations of the client.

### 3.4     Service negotiation

Once services have been discovered, there is a need to establish a relationship between the parties, (possibly mediated via third parties) and to negotiate the terms and conditions for the use of the service.

As part of this process, there is a step of trust evaluation, either from previous experience of one another, as recorded in a "trustbase" of trust valuations, or an evaluation of the trust value from recommendations from third parties, or a calculation of trust across the network via intermediate trust valuations. Preliminary work in calculating trust values across trust networks in the semantic web have been studied by Goldbeck, Hendler and Parsia [6; 7], and Richardson, Agrawal, and Domingos [14] which use a relatively straightforward model of trust which does not take into account context or uncertainty. They also consider reputation management, although this could be handled via an existing W3C recommendation, the Platform for Internet Content Selection (PICS [11]). This standard, designed originally for content filtering of web pages, can be used to express a general rating scheme, including for standards of reliability of web entities.

Once a trust valuation of the parties involved has been established an agreement needs to negotiated between the parties. This requires the interchange of vocabulary, and again the Ontology support provided via OWL in the Semantic Web is able to provide this mechanism; indeed as already noted, DAML-S has already started this for service descriptions, and KAoS for policies. We need to embed the trust valuations into the process, to use the expression of user requirements and preferences.

### 3.5     Experience monitoring and policy enforcement

Once an agreement has been established, then the client can start using the service. This usage may be long-lived, and the experience of the parties during the interaction may modify their behaviour for its remainder. For example, good experience may result in the loosening of restrictions and a higher-level of trust, changing the valuations in internal "trustbases", and reducing the policy enforcement overhead.

End-point enforcement of a web service in particular requires an agent:
- to interpret the data elements and procedure calls of messages, compare them with the rules in a policy statement and block unauthorized requests;
- to interpret outgoing messages in order to ensure that the service does not initiate communication with malicious agents or send unauthorized requests;
- to initiate an action or a specific request to another service in order to meet an obligation associated with the enforcement of a policy statement.

These processes are likely to take place largely internally to the services taking part, but nevertheless a common vocabulary for interpreting the action of the other and third parties is needed.

### 3.6 Service review

After the interaction has been completed, there should be a stage of review when trust valuations are reassessed, policies modified in the light of experience, and any recommendations to third parties propagated, within the common vocabularies provided within the Semantic Web.

## 4 Conclusion

Clearly, trust management, contract management and autonomic security mechanisms are important aspects in the practical deployment of the service architecture across different organisations, especially involving governmental and commercial organisations. Failure to provide adequate technical, legal and economic mechanisms to allow participants to act with confidence, will slow the acceptance the service architecture as an enabler of collaborations, and may prevent its uptake altogether, users instead using closed proprietary solutions which lack the benefits of an open system.

Semantic Web Services (e.g. http://www.swsi.org/) complement the rapidly advancing Web Services technology by defining and implementing new capabilities which more fully harness the power of Web Services through explicit representations of the semantics underlying Web resources. They provide an infrastructure capable of fully exploiting these semantics. Semantic Web Languages such as OWL extend RDF to allow users to specify ontologies composed of taxonomies of classes and inference rules. This is expected to allow software agents to understand and autonomously manipulate other agents or services, therefore enabling discovery, meaningful communication and collaboration among software agents and services, relying on control mechanisms that implement policy statements capturing human imposed constraints.

Thus the Semantic Web offers the infrastructure to share the vocabulary and semantic of policies and trust valuations. This has the advantage of using an established body of languages and tools designed to function over open distributed systems to leverage this sharing in an effective and economic manner. There has been some preliminary work in providing piece of this architecture carried out in different places. However, there has been no coherent scheme to bring these together in one policy-based service architecture, as outlined in this paper. In future work, we propose to provide a unified view to instantiate this architecture and consider how to deploy it in practice.

## Acknowledgements

# References

[1] Berners-Lee, T., Hendler, J., and Lassila, 0.: The Semantic Web. Scientific American, May 2001

[2] Bradshaw, J., Uszok, A., Jeffers, R., Suri, N., Hayes, P., Burstein, M., Acquisti, A., Benyo, B., Breedy, M., Carvalho, M., Diller, D., Johnson, M., Kulkarni, S., Lott, J., Sierhuis, M. and Van Hoof, R. Representation and reasoning about DAML-based policy and domain services in KAoS. In Proc. of The 2nd Int. Joint Conf. on Autonomous Agents and Multi Agent Systems (AAMAS2003).

[3 ] DAML Services homepage http://www.daml.org/services/

[4 ] Denker, G., Kagal, L., Finin, T., Paolucci, M. and Sycara, K. Security for DAML Web Services: Annotation and Matchmaking. In D. Fensel, K. Sycara, & J.Mylopoulos (Ed.), The Semantic Web—ISWC 2003. Proceedings of the 2nd International Semantic Web Conference, Sanibel Island, Florida, USA, October 2003, LNCS 2870.

[5] Dimitrakos, T., Matthews, B. and Bicarregui, J. Towards supporting security and trust management policies on the Web. ERCIM Workshop 'The Role of Trust in e-Business' in conjunction with IFIP I3E conference on October 3, 2001.

[6 ] Golbeck J., Parsia B., and Hendler J.: Trust networks on the semantic web. In Proceedings of Cooperative Intelligent Agents 2003, Helsinki, Finland, August 2003.

[7] Golbeck, J. and Hendler, J. Inferring Reputation on the Semantic Web. http://www.mindswap.org/papers/GolbeckWWW04.pdf, submitted to WWW'04.

[8] Johnson, M., Chang, P., Jeffers, R., Bradshaw, J. M., Soo, V.-W., Breedy, M. R., Bunch, L., Kulkarni, S., Lott, J., Suri, N., & Uszok, A. KAoS semantic policy and domain services: An application of DAML to Web services-based grid architectures. Proceedings of the AAMAS 03 Workshop on Web Services and Agent-Based Engineering. Melbourne, Australia, 2003.

[9] Kagal, K., Finin, T. and Anupam, J. A Logical Policy Language for a Pervasive Computing Environment., 4th IEEE Int. Workshop on Policies for Distributed Systems and Networks, Lake Como, 4-6 June, 2003.

[10] McBride, B., Wenning, R. and Cranor, L.: An RDF Schema for P3P. W3C Note 25 January 2002 http://www.w3.org/TR/p3p-rdfschema

[11] Platform for Internet Content Selection (PICS) http://www.w3.org/PICS

[12] Platform for Privacy Preferences (P3P) Project http://www.w3.org/P3P/

[13] Resource Description Framework homepage http://www.w3.org/RDF/

[14] Richardson, M., Agrawal, R. and Domingos, P. Trust Management and the Semantic Web In D. Fensel, K. Sycara, & J.Mylopoulos (Eds.), The Semantic Web—ISWC 2003. Proc. of the 2nd Int. Semantic Web Conf., Sanibel Island, Florida, USA, October 2003, LNCS 2870.

[15] Tonti, G., Bradshaw, J. M., Jeffers, R., Montanari, R., Suri, N., & Uszok, A. (2003). Semantic Web languages for policy representation and reasoning: A comparison of KAoS, Rei, and Ponder. In D. Fensel, K. Sycara, & J.Mylopoulos (Eds.), The Semantic Web—ISWC 2003. Proc. of the 2nd Int. Semantic Web Conf., Sanibel Island, Florida, USA, October 2003, LNCS 2870.

[16] Web-Ontology (WebOnt) Working Group homepage http://www.w3.org/2001/sw/WebOnt/

[17] Uszok, A., Bradshaw, J., Jeffers, R., Suri, N., Hayes, P., Breedy, M., Bunch, L., Johnson, M., Kulkarni, S. and Lott, J. (2003). KAoS Policy and Domain Services: Toward a Description-Logic Approach to Policy Representation, Deconfliction and Enforcement. In Proc. of the IEEE Workshop on Policy 2003.

[18] Uszok, A., Bradshaw, J., Jeffers, R., Johnson, M., Tate, A., Dalton, J. and Aitken, S. Policy and Contract Management for Semantic Web Services. to appear AAAI Spring Symposium, Stanford University, California, USA, March 2004.

[19] XACML: Extensible Access Control Markup Language

http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml