

Policy Driven Composition of Trustworthy Web Services

Karl Quinn, Declan O'Sullivan, Vincent Wade

Knowledge and Data Engineering Group

Trinity College Dublin

1.0 Introduction

Definitions of trust generally use synonyms or trust inspiring terms in their definition. “Belief” [McKnight96], “Credibility or Reliability” [Golbeck03], “Confidence or Faith” [Shadbolt02], “Reputation” [Golbeck04], “Competence and Honesty” [Grandison00] have all been used in this way. Definitions generally try to convey that trust has a specific, quantitative, and directed (i.e. A to B, not B to A) value. Increasingly the definition of trust values and their calculation are seen as important elements of an overall security framework. Determination of an end-to-end trust value for a particular service can be brought about by reasoning over the trust metadata (including trust values) associated with each of the individual service components. Web Services will benefit from the use of trust metadata and management as it can aid in the automatic discovery or composition of trustworthy web services.

In parallel, a major direction of web service research is towards service collaboration and semantic annotation [Zhang03] through ontologies. Ontology provides a means to describe and define terms, concepts, and relationships specific to a knowledge domain such as trust and services. The development of technologies for the Semantic Web [BernersLee01] has produced ontology languages such as the W3C's OWL (Ontology Web Language) [McGuinness03] which can be reasoned over at runtime. It is argued that trust and service metadata described with reference to such ontologies provide for improved reasoning because of its semantic basis. Trust management research is also exploring models for managing trust on an internet scale outside of the more traditional centralized systems. Furthermore trust management research has also exploring the use of policy based management [Bfix98, Bfl96, Bfs98, Keromytis03].

Given the above directions, our research focuses on combining ontology and policy based approaches for trust management. This involves the annotation of web services with trust metadata with reference to ontologies for Trust and Services which is under development. In addition users will be enabled to specify policies that determine how the management system should reason over the trust metadata for trustworthy service selection and composition.

This extended abstract: provides an overview of the key concepts of semantic web services, policy and trust; highlights important related work; and introduces our current experimentation in the area. The full paper will provide details of the results and experiences arising out of our current experiment (due for completion August 2004).

2.0 Key Concepts

Semantic Web Services

Web Services can be defined as “self-contained, modular applications that can be described, published, located, and invoked over a network—generally, the World Wide Web” [Gottschalk]. Web Services enable businesses, governments and individuals to easily integrate many of their applications that may be from heterogeneous environments, which can be both internal and external to the organization.

However a significant difficulty with the current web is that the majority of the resources have been designed for use by humans primarily. The idea behind of the Semantic Web initiative is to provide tools and techniques to allow resources on the web to be augmented with information that would allow for greater interpretation and processing by computer applications directly.

Semantic Web Services are the combination of Web Services and Semantic Web. In this approach, ontology languages from the Semantic Web community are used to describe the properties and capabilities of web services in an unambiguous, machine-interpretable form. Currently, the composition of web services relies more on manual hard-coding than on automated service orchestration. The promise of Semantic Web Services is that they allow for the automation of web service discovery, invocation, interoperation, composition and execution monitoring. Early research has shown [Ruoyan03, Narayanan02] and confirmed [Srivastava03, McIlraith01, McIlraith03] the availability of these kinds of automation.

Trust

Trust has various definitions that are applicable to different areas of computer security. Many definition use synonyms or trust inspiring terms in their definition. “Belief” [McKnight96], “Credibility or Reliability” [Golbeck03], “Confidence or Faith” [Shadbolt02], “Reputation” [Golbeck04], “Competence and Honesty” [Grandison00] have all been used in this way. Definitions generally try to convey that trust has some quantitative value associated with it such that A trusts B, but only by so much. Trust is multidirectional in that B may not trust A at all. Trust can be made specific even more by stating that A trusts B in relation to car maintenance but not with regards to medical procedures. Trust and its synonyms can be applied to many facets of the A and B relationship. It can be interpreted that A trusts the information that B (BBC News service) provides if A finds that the BBC’s information is both credible and reliable. From this A can assign some level of confidence to the BBC’s information and act in good faith upon it. The BBC can build up its reputation (expectation of behaviour based on past observations/information) by using competent reporters so that A has a directed and weighted belief in the BBC and its statements. The BBC can at the same time hold (to some degree) confidence in its audience.

Trust is a difficult issue to contemplate because it is such a human idea that has so many uses and general meanings. It is for this reason that the scoping of trust (much like security) can enable a more specific meaning that can be more efficiently translated into mission goals and statements. For the purposes of our research trust can be seen as the aggregation of many of the synonyms used above in conjunction with the ideals that they convey. Web Services must be reliable and inspire confidence in their users. Their information must be credible (or from credible sources) and honest so that it can be believed. Together all these elements can create a reputation of trust that can help in such areas as the orchestration of composite Web Services. In the real world we tend to make choices about who we buy services like health insurance from based on trust because it gives us peace of mind, so it seems plausible that we should have similar notions of trust in the computing world in order to help make choices about online service composition.

Policy

A policy can be seen as a form of rule, which may change the behaviour of an entity. Generally, a policy is expressed in the form of the triple: Event; Condition; Actions, where the event triggers an evaluation of the policy rule. The conditions are a set of stipulations that must be met in order for the policy to be enacted. If a policy is to be enacted then the actions state what is to be performed. Policy languages are typically vendor specific and proprietary [Carey03]. In general, policy languages are split between access control and resource management languages [Sloman02].

Policies are useful for applying a common set of rules to a large set of distributed nodes, services or users. Policies capture the business goals of an organization, which will allow these policies to govern how the organization's or individual's system operates. REFEREE [Chu97] is a system that allows users to specify rules as policies in order to provide trust management for web applications. [Kagal04] uses policies to specify privacy rules for selecting web services.

3.0 Related Work

Semantic Web Services & Trust

The creation and utilization of domain specific ontologies, such as a trust ontology, on the Semantic Web will empower a richer semantic environment in which more powerful expression and reasoning may occur. Semantic Web Services will take advantage of ontologies by using them in their advertisements and so enable more sophisticated and accurate service discovery and matching to occur.

There are a limited number of ontologies that are already defined for trust and reputation (a sub element of trust). In [Golbeck04] an extension is made to the Friend-Of-A-Friend (FOAF) ontology [FOAFont] that allows for the assignment of a reputation value to a person. This extension to allow for the reputation value is similar to Golbeck's earlier work [Golbeck03] where the value assigned was for trust. Both [Golbeck03] and [Golbeck04] describe how trust/reputation can be applied to a person for a specific subject area. For example, Bob can state that he trusts Dan in relation to a certain area by such a degree or that Dan has a certain reputation in a specific area. The levels of trust used by Golbeck are defined at <http://trust.mindswap.org/ont/trust.owl> where trust values are measured on a scale from one to ten, one being absolute distrust and ten being absolute trust.

Ontologies can be used by systems in order to express and calculate trust. The TRELIS [Yolunda02] project purports to enable users to express their trust in a source (and the sources statements) so that an individual's trust can be combined into an overall assessment of trust. It is presented as an information analysis tool that enables users to annotate how they analyze and use information when making some decision. Friend-Of-A-Friend (FOAF) [Dumbill02] is a project that has a RDF vocabulary that users can use to describe information about herself and her friends, which includes statements that can be used to build a web of acquaintances. A users' privacy is optionally maintained by use of signed files that specify anonymity or their true identity. The Advogato [Levin98] project also automatically calculates trust using group assertions. Interestingly, the Advogato metric for calculating trust is highly attack resistant. This allows the system to cut out portions of the network that are subsequently identified as 'bad'. Advogato, FOAF and TRELIS all provide a platform for the basis of a trust network that is commonly referred to as a 'Web of Trust', which is one of the ultimate goals of the Semantic Web.

Trust management systems for traditional computing environments such as PolicyMaker [Bfl96, Bfs98] and KeyNote [Bfik98] are two (similar) engines for granting authorization. Instead of the two step process of authentication and access control for processing a (signed) request these engines address the authorization problem directly. In the two step process the questions asked are "Is this person who they say they are?" and "does this person have the correct access control permissions for this request?". Even the reliable authentication of a user 'a priori' wouldn't help in deciding whether or not to execute the requested action of a user, if the user is unknown. Therefore, in directly

answering the authorization problem the question asked becomes “is the key that signed this request authorized to take this action?” or as the [Bfl96] approach puts it “Does the set C of credentials prove that the request complies with the local security policy P?”. Any entity that responds to requests must have a (local) policy that acts as the ultimate source of authority on which decisions may be directly based upon. The policy can delegate this responsibility to credential issuers that it trusts and that have the required domain expertise as well as relationships with potential requesters. This delegation process enables entities to grant authorization to users’ that it doesn’t know ‘a priori’. KeyNote builds upon PolicyMaker by adding two design goals; standardization and easy integration into applications.

Vigil [Kagal02] is a trust management system for pervasive computing environments that uses an ontological approach for the development of a Role Based Access Control like system. According to this research an ontological approach enables the system to extend trust based on the user’s role, where roles can be changed based on a users actions or context. For example someone in a meeting room using the projector can be assumed to be the speaker and therefore the use of the computer could be granted to them based on their speaker context. The ‘Security Agent’, among other things, manages trust and receives new or altered access rights information and enforces policies in the local space. A system of delegation is used to allow users with no access rights to access a particular resource so long as a user who has the correct access rights delegates this ability. A delegation system like this could be used to handle unknown users a priori so long as a known user delegates the appropriate rights.

Semantic Web Services & Policy

Three common policy languages are Ponder [Damianou00], Rei [Kagal03], and KAoS [Bradshaw97]. Each policy language has its own model and language specification.

PONDER is an object oriented policy language for several types of management policies for distributed systems and also provides techniques for policy administration. There are basic policies and composite policies where composite policies are simply groups of basic policies. For example the ‘role’ composite policy can govern the behaviour of a subject by specifying its rights and duties. The basic policies themselves are specified by declarations and between sets of subjects (users) and sets of targets (resources). There are two main policy types; obligations and authorizations. Obligations are actions that a subject must perform on a target object when a specific event occurs. Authorizations are operations that a subject is allowed perform on a target object.

Rei is a concept based policy language that allows for the specification, analysis, and reasoning of policies. It enables users to express and represent the concepts of right and prohibitions (positive and negative authorizations) as well as obligations and dispensations. The concepts are represented in an application independent ontology that allows for greater interoperability with other policy languages and also enables user to extend the ontology as required. A Rei policy is a rule that associates an entity with its set of rights, prohibitions, obligations, and dispensations. An action, as outlined by its

ontological description, is the composition of an identity, target object, pre-conditions and effects. Rei has a set of speech acts primitives that allows the system to exchange rights and obligations between entities. Meta-data is used to resolve policy conflicts that the Rei policy engine encounters.

KAoS is a DAML/OWL policy language that is a collection of componentized policy and domain management services for (among other things) web services. Domain services allow entities to be categorized into domains to enable e-commerce collaboration and external policy administration. Policy services empower users to specify, manage, resolve conflicts and enforce policies within domains. KAOs uses authorizations and obligations as part of its policy ontology. Components can be annotated regardless of whether the component was designed for it in advance. KAOs supports run-time policy changes and is extensible to a variety of platforms.

Policies that are defined in ontology languages could be considered to have a level of semantic richness that makes for easier policy conflict detection and resolution and for easier integration with Semantic Web Services. The very nature of ontologies would suggest that ontology based policy languages, like Rei, would provide greater interoperability with policies defined in other languages.

Security & Policy

Policies that specify trust rules have also been investigated. Blaze's PolicyMaker and its successor KeyNote have used policies for describing and implementing trust relationships in order to be granted authorization to perform specific trusted actions.

Kagal has combined the separate research areas of an ontology based policy language with ontology based semantic web services. [Kagal04] describes how privacy policies could be written as Rei policies that are then used during the discovery phase of web service selection. For example, a person can specify that in order for their communication to be private the channel in which the messages propagate must be encrypted. A services description could then be used to ascertain whether that service can provide the necessary secure channel to meet the requirements of the privacy policy. So long as a candidate service does not violate the users' privacy policies it may be selected for use.

4.0 Experimentation

The aim of our current experimentation is the development of a platform that will allow Semantic Web Service composition based on trust annotations and where policies will provide decentralized management for a user's trust requirements. The key challenges (as presented in Diagram 4.1) involved in this research are: development of ontology specific to trust and services, annotation of selected web services according to the ontologies, and reasoning over the annotated services to establish end to end trust values.

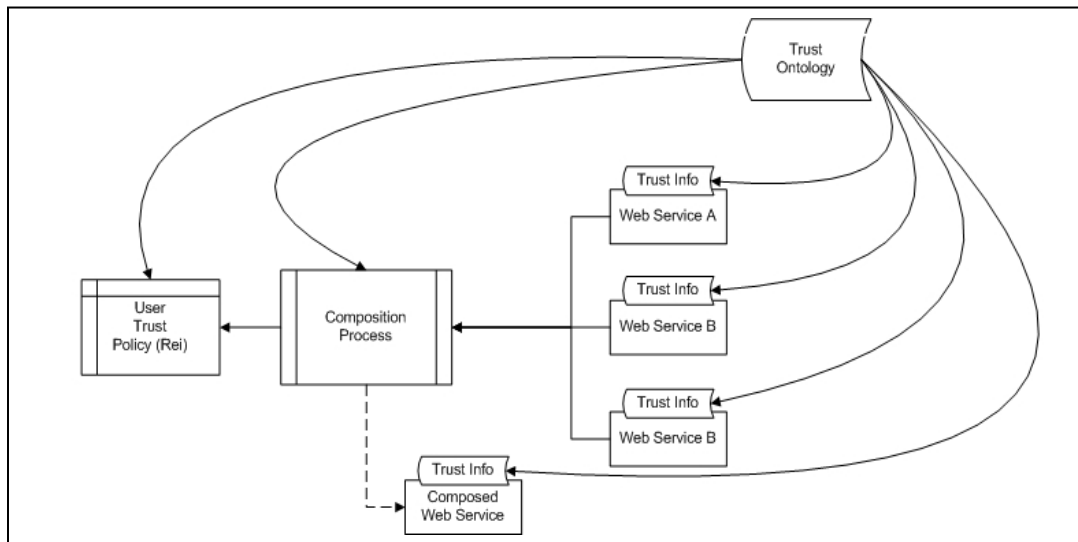


Diagram 4.1: High Level Architecture

The current state of the art in trust ontologies is rather elementary with only the very basic concepts of trust and sub-elements of trust actually developed. A definitive (or even semi-complete) trust ontology or trust schema (trust mappings) has not yet been defined. In a similar vein to Golbeck, but at a more complete level, a trust ontology is currently being designed (using the Web Ontology Language OWL) to support the semantic annotation of web services in a fashion similar to Kagal's research efforts. Yet where Kagal offers privacy as a determining factor in the selection of a web service our experiment is using trust in the selection and composition of web services. Like Blaze, Kagal, and the W3C policies are being used (using Rei) to describe security requirements, specifically trust, in order to locally enforce user defined rules.

The full paper will describe results and experience drawn from the initial experiment (due for completion in August 2004) with respect to the trust ontology; the annotation of the web services and reasoning about trust information in order to orchestrate web service composition.

References

[Bfl96] M. Blaze, J. Feigenbaum, and J. Lacy. Decentralized Trust Management. In Proc. of the 17th Symposium on Security and Privacy, pages 164{173. IEEE Computer Society Press, Los Alamitos, 1996.

[Bfix98] M. Blaze, J. Feigenbaum, J. Ioannidis, and A. Keromytis. The KeyNote Trust-Management System. Work in Progress, <http://www.cis.upenn.edu/~angelos/keynote.html> , June 1998.

[Bfs98] M. Blaze, J. Feigenbaum, and M. Strauss. Compliance Checking in the PolicyMaker Trust-Management System. In Proc. of the Financial Cryptography '98, Lecture Notes in Computer Science, vol.1465, pages 254{274. Springer, Berlin, 1998.

[Bradshaw97] Bradshaw, J. M., Dutfield, S., Benoit, P., and Woolley, J. D., 'KAoS: Toward an industrial-strength generic agent architecture', In J. M. Bradshaw (Ed.), Software Agents. (pp. 375-418). Cambridge, MA: AAAI Press/The MIT Press.

[Cahill03] Cahill, V., et al, 'Using Trust for Secure Collaboration in Uncertain Environments', IEEE Pervasive Computing Magazine, special issue Dealing with Uncertainty, Volume 2, Number 3, pp – 52-61, July - September 2003.

[Carey03] Carey, K., Feeney, K., Lewis, D., 'State of the Art: Policy Techniques for Adaptive Management of Smart Spaces', M-Zones Deliverable 1, June 2003, http://www.m-zones.org/deliverables/D1_1/Policy.pdf

[Chu97] Chu, Y., Feigenbaum, J., LaMacchia, B., Resnick, P., and Strauss, Ma., 'REFEREE: Trust Management for Web Applications.', The World Wide Web Journal, 1997, 2(3), pp. 127-139.

[Damianou00] Damianou, N., Dulay, N., Lupu, E. C., and Sloman, M., 'Ponder: A Language for Specifying Security and Management Policies for Distributed Systems (Version 2.3)', Imperial College of Science, Technology and Medicine, Department of Computing, 20 October 2000.

[Dumbill02] Dumbill, E., 'XML Watch: Finding friends with XML and RDF.', IBM Developer Works, <http://www-106.ibm.com/developerworks/xml/library/xfoaf.html>, June 2002.

[FOAFont] RDFWeb: FOAF: 'the friend of a friend vocabulary', <http://rdfweb.org/foaf/>

[Golbeck03] Golbeck, J., Hendler, J., Parsia, B. 'Trust Networks on the Semantic Web', 12th International Web Conference (WWW03), Budapest, Hungary, May 2003.

[Golbeck04] Golbeck, J., Hendler, J., 'Inferring Reputation on the Semantic Web', 13th International Web Conference (WWW2004), New York, NY, USA, May 2004.

[Gottschalk] Gottschalk, K., et al, 'Web Services Architecture Overview: The Next Stage of Evolution for E-Business', <http://www-106.ibm.com/developerworks/library/w-ovr>
[Grandison00] Grandison, T., Sloman, M., 'A Survey of Trust in Internet Applications', IEEE Communications Surveys, 3, pp. 2-16, Fourth Quarter 2000.

[Kagal02] Kagal, L., Undercoffer, J., Perich, F., Joshi, A, Finin, T., 'A Security Architecture Based on Trust Management for Pervasive Computing Systems', Proceedings of Grace Hopper Celebration of Women in Computing 2002.

[Kagal03] Kagal, L., Finin, T., Joshi, A., 'A Policy Language for a Pervasive Computing Environment', Proceedings of IEEE 4th International Workshop on Policies for Distributed Systems and Networks (POLICY 2003), Lake Como, Italy.

[Kagal04] Kagal, L., Paoucci, M., Srinivasan, N., Denker, G., Finin, T., and Sycara, K., 'Authorization and Privacy for Semantic Web Services', AAAI 2004 Spring Symposium on Semantic Web Services, March 22, 2004.

[Keromytis03] Keromytis, A., Ioannidis, S., Greenwald, M.B., Smith, J.M., 'The STRONGMAN Architecture', In the *Third DARPA Information Survivability Conference and Exposition (DISCEX III)*, Washington, D.C. April 22-24, 2003.

[Levin98] Levien, R., Aiken, A., 'Attack resistant trust metrics for public key certification.', 7th USENIX Security Symposium, San Antonio, Texas, January 1998.

[McGuinness03] McGuinness, D.L., van Harmelen, F., 'OWL Web Ontology Language Overview', W3C Proposed Recommendation, 15th Dec 2003.

[McKnight96] McKnight, H.D., Chervany, N.L., 'The Meanings of Trust; Technical Report 94-04, Carlson School of Management, University of Minnesota', 1996.

[McIlraith01] McIlraith, S., Son, T.C., Zeng, H., 'Semantic web services', IEEE Intelligent Systems, 16(2):46-53, March/April 2001.

[McIlraith03] McIlraith, S. and Martin, D., 'Bringing Semantics to Web Services', IEEE Intelligent Systems, 18(1):90--93, January/February, 2003.

[Narayanan02] Narayanan, S., McIlraith, S., 'Simulation, Verification and Automated Composition of Web Services', WWW2002, May 7-11, 2002, Honolulu, Hawaii, USA.

[Ruoyan03] Ruoyan, Z., Arpinar, B., Aleman-Meza, B., 'Automatic Composition of Semantic Web Services', The 2003 International Conference on Web Services (ICWS'03), June 2003.

[Shadbolt02] Shadbolt, N., 'A Matter of Trust', IEEE Intelligent Systems, pp. 2-3 January/February 2002.

[Sloman02] Sloman, M., Lupu, E., (2002) 'Security and Management Policy Specification', IEEE Network, vol.16 No. 2 pp.10-19, March/April 2002.

[Srivastava03] Srivastava, B., Koehler, J., 'Web Service Composition - Current Solutions and Open Problems.', ICAPS 2003, Workshop on Planning for Web Services 10 June 2003, Trento, Italy.

[Yolanda02] Gil, Y., Ratnakar, V., 'Trusting Information Sources One Citizen at a Time.', Proceedings of the First International Semantic Web Conference (ISWC), Sardinia, Italy, June 2002.

[Zhang03] Zhang, L. J., Jeckle, M., 'The Next Big Thing: Web Services Collaboration', Proceedings of ICWS-Europe 2003, Springer, LNCS 2853, September 2003, pp 1-10.