Semantic Web Trust: The next step in web evolution

Steven Caruana and Matthew Montebello

Department of Artificial Intelligence, University of Malta
{scar009|matthew.montebello}@um.edu.mt

Abstract. The inception of the World Wide Web marked the beginning of a new age, an age where information is easily distributed and accessible to everyone. Years after it was conceived, the net is still growing at a higher rate than ever. It is becoming ever more apparent that the World Wide Web's current software infrastructure will need to evolve if it is to remain a reliable and dependable resource. In this paper we will be looking into the facets that make web content accessible and reliable. We will also be proposing a structure that makes use of technologies such as the Semantic Web and Agent Technology to help resolve data access and classification issues. The approach that we are proposing will involve the creation and distribution of data tags, policies and reasoners. We will also show how these items can be used by entities such as software agents or users to specify access control, classify the resources in terms of relevance and to decide on how trustworthy the information being used is.

1 Introduction

When a platform is released to the general public, it is inevitable that users will try to break it down. The internet is no exception to this rule and has already sustained a very wide number of different attacks. During the last few years both industry and academics have embraced these issues and have been developing successful solutions to each of the identified threats.

Until a few years ago, the idea of storing sensitive information such as credit cards or personal documents on the web would have sounded like a fairly ambitious goal. When this was first attempted there were numerous cases of people who had their details stolen and maliciously used both on and off the net. Nowadays users will not only trust the web with this information but will also rely on it to perform most of the functions that would usually be done by the person manually.

With security threats being ironed out regularly, it is now imperative that we start to look towards the next step in the information superhighway's evolution. The idea is to start shifting the focus of web research from the storage and transfer of information to trying to make it more valuable to the user.

In the following sections we shall be discussing the motivations for needing a new extension to the net and shortly after is a description of a number of structures that are aimed at making this system a reality.

2 Security and trust

In this paper a distinction is made between security and trust. Security is the term used to refer to how prone information is to being stolen or misused. On the other hand, trust is the term we use to represent confidence that users have in the data they are accessing. Our view of trust falls in with Cristiano Castelfranchi's and Rino Falcone's [CF00] view of trust. They stated that trusting an entity involves both the acceptance of a certain amount of risk and that this risk should be assessed and quantified. In their paper they also stated that only a cognitive agent can decide whether to trust another agent or not.
The definition of a cognitive agent is that of an agent which requires both goals and beliefs. The current web does not enforce a standard to define these structures. For agents and users alike to be able to decide intelligently on whether a piece of information can be trusted, we require a new structure. This structure will need to handle both the specification of such notions and to make provisions for the resolution of such dilemmas.
In the last few years there have been numerous advancements in the field of security but little progress was made in trying to make data trustworthy. There are numerous factors that can determine how trustworthy data is. We envisage that the WWW requires a structure that can represent such relations so that it can in turn be made available to users to help them decide on how essential and relevant what they are viewing is to them.

3 Looking beyond authentication

Standard authentication provides users with a means of asserting their identity on the net but makes no provisions for users to express relations associated to the resource.
Relations are important because they allow users to customise complex structures that can be used to express information relating to the data. Once defined, these relations can then be used by the servers for access control or by users for filtering out what they are really interested in viewing.
There have already been a number of proposals for systems that deal with trust resolution in social networks. [MA04] proposes a solution for the calculation of the trust level attributed to an item. These calculations are based on the feedback set by users of the collaborative system. This structure is but one of the methodologies that can be used to express trust. In [VDCP06] the authors discuss how the calculation of trust can be done using different methodologies and how these methods can be applied to different scenarios.
In the papers mentioned above, all the systems that were referred to made use of boolean quantifiers to provide results for the resolution of trust evaluation. In [VDCP06] the authors also discuss trust functions that, rather than returning a boolean quantifier return a score that will signify how reputable the source is. [WV03,Mui03] explore the possibilities of reputation systems to decide on the reputation attributed to objects in different scenarios.

In the following section we will be looking at a structure that aims at providing an extension for the web that will allow us to add trust to the World Wide Web.

4 Introducing Trust into the Web

While designing this project we realised that it would be necessary to ground our work using Semantic Web technology as one of our building blocks. Semantic Web implementations of trust resolution frameworks have been under development for a number of years now and projects such as Rein [KBL05] and KAoS [UBJJ04] have already released implementations of current drafts.
As mentioned earlier, the system we are developing requires that entities, relations and rules be described in a machine and human readable language. DAML+OIL [Hor02] and OWL [DCHH+02] are two languages born from the Semantic Web initiative. The layer responsible for the trust reasoning will be composed of two subsections. The first section will constitute an interface into which a reasoning engine can be plugged in. The other part of this repository will refer to an archive that will be used for the storage.

4.1 Semantic Web

To construct this framework we chose the Semantic Web to be one of the building blocks. What made the Semantic Web a plausible foundation is the fact that it provides languages and tools that can be used for the tagging of information. When the Semantic Web [BLHL01] was first released to the general public, trust was defined to be one of the problems that the Semantic Web Research aimed at resolving. This project goes very much hand in hand with this initiative.
The reason for us adopting these technologies is the fact that the Semantic Web provides us with a set of languages that have developed specifically for the describing of entities and the relations between them. Languages such as OWL [DCHH+02] are already being used to build social networks such as Foaf [FOAF].

4.2 Agents

Agents play a very central role in this setup. Automation of processes has always been a very important aspect of computing. To date computers are already used to automate known processes. This implies that if a process can be formalised, then applications can be written to relieve the user of the repetitive parts of their jobs.
We believe that the next step in the design of automation software is that of creating applications that can make decisions in environments that they have not been designed to deal with. The development of trust layers is envisaged to provide agents with a set of tools that will help them deal with such scenarios.
If users are given a means to define what information can be trusted, then they can also specify what resources agents are to trust and how to make use of

them. It is our belief that reasoning layers are not to be used exclusively by web applications, but also by applications such as agents to define the flow that can be used to perform certain tasks.
Another important role that we foresee agents will be taking on, is that of access control to information. Different users have different needs. The system that we designed is meant to help users define these needs and to make it possible to express them to agents. Agents can be used to filter information on the net that does not fall within the criteria that the user has specified.

5 Infrastructure Design

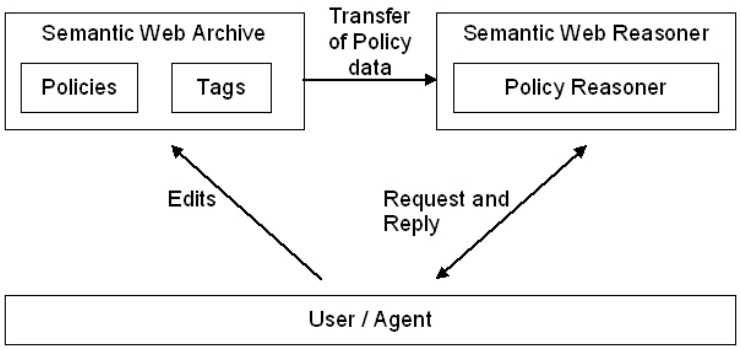

Fig. 1: The diagram above depicts the two main modules in our design and their respective internal structures. The lines between them show the interaction that takes place between these modules and the users of the system.

The design adopted by this project is a modular one. In modularising the components of this system we believe that it will be possible to attain a number of loosely coupled structures that can interconnect at runtime and be deployed in different locations and on different platforms.
The platform we designed is made up of two major modules which are the Semantic Web Archive and the Semantic Web Reasoner. Semantic Web Archive modules can be deployed at different locations and will store the tags that users will define. Like Semantic Web Archives, the Semantic Web Reasoner modules can also be deployed at different locations. Semantic Web Reasoner modules are responsible for the storage of the reasoners that will be used in this semantic network.

5.1 Semantic Web Archive

The Semantic Web Archive is the module that will be responsible for the storage of both the policies and tags. In a typical setup, instances of these modules will

be spread across the net and users who have access to them will be offered an interface through which they can manage the archiving of policies and tags.
This module is to be used by both users and reasoners. Users will be provided with an interface hosted directly through the module. This interface will help explain the policies that are stored in this module and will also provide visual aids for the user to annotate the data that is stored on the web server.
Semantic Web Reasoners on the other hand will be provided with an interface that will allow them to query this module for the resources it is hosting. When a query is received, this module will first search for the data that is being requested and will then marshal it via a web service back to the Semantic Web Reasoner.
For security reasons we believe that this module will require a guard. If unauthorised users or applications were to gain access to this module, they could download the policies and tags and find ways of exploiting them. The guard's main goal will be responsible for both the authentication of the reasoners and that of the users. For users, a standard log-on mechanism is envisaged to provide the access mechanisms necessary to authenticate them. On the other hand, Semantic Web Reasoners will need to be registered with this module before they can be allowed to query the system. The registering of Semantic Web Reasoners will be done by administrators managing these modules.

5.2 Semantic Web Reasoner

The Semantic Web Reasoner module is made up of two layers. These are the communication layer and the reasoning engine layer. The reasoning engine was designed to support pluggable reasoning modules. Developers can develop their own reasoners and embed them into the Semantic Web Reasoner.
The communication layer of this module will handle the communication between services connecting to the Semantic Web Reasoner and the interconnections between the Semantic Web Reasoner and the Semantic Web Archive modules. This layer will also allow for calls to be generated between other external heterogeneous modules to process parts of these requests.
Web applications will use this module to determine the result of policy requests. Based on the feedback returned by the Semantic Web Reasoner, the applications will be able to classify the resource and decide on how to approach it.
Like Semantic Web Archives, even Semantic Web Reasoners were designed to have a guard mechanism. In this case the guard is there for the registering of Semantic Web Reasoners between each other. This will ensure that when being called by another Semantic Web Reasoner, the calls are genuine.

5.3 Tags and Policies

The system being proposed is based heavily on the generation and parsing of tags and policies. It is for this reason that we believe that it is imperative that the process of creating tags and policies be simple whilst still powerful enough to express the necessary relations.

To address these issues we designed a system that relies mostly on web forms to provide users with a standard implementation of an editor that can be accessed from different platforms. The current web is geared at providing a rich user experience. Our design involves the use of a graphical user interface that will be responsible for the visualisation of data structures (tags) and policy pipelines.
Policies stored in Semantic Web Archives are allowed to span across resources and policies that are found on remote servers. It is for this reason that if a policy is to be reasoned out properly, the Semantic Web Reasoner will need to be registered with all of the required modules.

5.4 Policy Reasoning

The approach we decided to take in policy reasoning is that of creating a virtual structure resembling a pipeline. These pipelines are to be defined by users and each pipeline is to be made up of modules. In turn, each module is to receive a set of inputs and will generate a set of outputs. Once an output is produced, it is then pumped back into the next section of the pipeline. At the end of the pipeline, a set of results will be generated and sent back to the user or service. It will then be up to the system to decide how to react to this information.
Status propagation is another important aspect of this structure. If a user runs a query that fails whilst executing (e.g. a Semantic Web Reasoner or Semantic Web Archive is not found), then the user will need to be notified of this issue. To do this we decided that at every hop the call makes, an xml message will be generated and at every call more logs will be appended to it. When a call terminates (both successfully or in an error), this message is returned to the user's machine or agent and the appropriate action is performed.

5.5 Publishing of Policies and Tags

Distribution of information is one of the major concerns of this infrastructure. If the data is not published properly, the consequences could be twofold. The data could either be too well hidden to be accessible by users who might need to reference it, or it could be accessible to the point where users with malicious intents could abuse of it.
To solve this problem we looked again at the web's current infrastructure. We believe that the safest structure for this kind of network is that of having a secured network of data access points that can rely on existing standards of data encryption and security. For a user to have access to the various Semantic Web Archive and Semantic Web Reasoner nodes, they will need to have user accounts that are to be created by the administrators or service owners as discussed in the previous sections.
This structure will allow anyone deploying their policy network to decide for themselves how public they can afford their networks to be. Once the networks have been established, a user base can be maintained using the same structures that are being proposed in this paper. Servers can be made to decide which users will have access to the resources it is publishing.

5.6 Adoption of Relation Tagging

Making the shift to relational data on the net has been part of the focus of this project since its inception. It is for this reason that part of this project is aimed at providing user friendly tools to help better understand the potential uses of relations and to make them accessible to as wide an audience as possible.
It is a known fact that if users are presented with an unfamiliar environment they will feel reluctant to adopting new mechanisms. To address this problem, we envisage the adoption of a graphical user interface that can be called up from a browser. This interface is to be accessible from anywhere on the net and ubiquitous across most platforms.

6 Conclusions and Future work

The addition of trust to the web infrastructure is an important step in the web's evolution. As the web grows, we are slowly losing control over the authenticity of data published and trust in web applications is a topic that will become ever more important as time goes by.
Once this framework is constructed, we believe that more work will be necessary to ensure that the framework is compatible with different environments (such as mobile phones) that might not support the tools that we are offering.

References

[MA04] P. Massa, P. Avesani.: Trust-Aware Collaborative Filtering for Recommender Systems. Springer-Verlag GmbH, 3290, 2004

[VDCP06] P. Victor, M. De Cock, C. Cornelis, P. Pinheiro da Silva.: Towards a Provenance-Preserving Trust Model in Agent Networks. Proceedings of Models of Trust for the Web (MTW'06), Workshop at the 15th International World Wide Web Conference (WWW2006), ISBN 085432853X, 2006

[WV03] Y. Wang, J. Vassileva.: Trust and Reputation Model in Peer-to-Peer Networks. p2p, p. 150, Third International Conference on Peer-to-Peer Computing (P2P'03), 2003

[Mui03] L. Mui.: Computational Models of Trust and Reputation: Agents, Evolutionary Games, and Social Networks. Ph.D. Dissertation, Massachusetts Institute of Technology. 2003

[KBL05] L. Kagal and T. Berners-Lee.: Where policies meet rules in the semantic web. Technical report, MIT, 2005

[UBJJ04] A. Uszok, J. M. Bradshaw, M. Johnson, and R. Jeffers.: Kaos policy management for semantic web services. IEEE INTELLIGENTSYSTEMS, 284(5):3241, July-August 2004.

[Hor02] I. Horrocks.: DAML+OIL: a reason-able web ontology language. in Proceedings of EDBT 2002, March 2002.

[DCHH+02] M. Dean, D. Connolly, F. Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider, L. Andrea Stein.: OWL Web Ontology Language 1.0 Reference. http://www.w3.org/TR/2002/WD-owl-ref-20020729. July 2002

[FOAF] The Friend of a Friend (FOAF) project.: http://www.foaf-project.org/

[BLHL01] T. Berners-Lee, J. Hendler, and O. Lassila. : The Semantic Web. Scientific American, 284(5):34–43, 2001.

[CF00] C. Castelfranchi and R. Falcone. : Trust Is Much More than Subjective Probability: Mental Components and Sources of Trust. IEEE Computer Society. Mental Components and Sources of Trust. HICSS 2000