Enhancing P3P Framework through Policies and Trust *

Pranam Kolari, Li Ding, Lalana Kagal, Shashidhara Ganjugunte
Anupam Joshi, Tim Finin

University of Maryland Baltimore County, Baltimore, MD 21250, USA

Abstract. The Platform for Privacy Preferences (P3P) is a W3C standard that websites can use to describe their privacy practices. The presence of P3P policies enable users to configure web browsers to constrain what they can and cannot do when visiting websites. It's a good idea that unfortunately is rarely used. We identify two reasons: (i) the languages available to define a user's privacy preferences are not very expressive and (ii) most websites do not have published P3P policies. We present enhancements to P3P framework that uses trust and the Semantic Web concepts to solve these problems. We use the RDF-based Rei policy language to enable users to describe their privacy-related constraints and preferences. Further, our approach is effective even in the absence of published P3P policies through the incorporation of our trust model. We present use cases to demonstrate the relevance of our work to the current web privacy landscape and offer it as a powerful enhancement that can promote P3P's adoption and use.

1 Introduction

The issue of "web privacy" is increasingly important to users. While accessing an online vendor or even just browsing a website, users' private information is often collected explicitly or implicitly for tracking or targeting. An example of implicit information is the click-stream, which is the sequence of pages visited by a user. Many websites also require users to register and provide personal data like name and contact details. Moreover, distributed data mining [9] and other techniques can track a user across websites, user sessions and physical locations. Hence it is very important for users to be aware of the potential privacy hazards and have better control over disclosing private information in their online activities.

The first step towards protecting web privacy is for a user to read the privacy policies published by websites and then decide how to interact with them. This is often the only step available today. Manual perusal is however a time consuming task and not practical for a user who visits tens of websites each day. Motivated by this manual limitation, W3C proposed the P3P framework (http://www.w3.org/P3P/) for automating the privacy policy verification process. The P3P framework (i) requires that websites publish XML based privacy policies using the P3P vocabulary, (ii) lets users specify their privacy preference

* This work was supported in part by NSF grants IIS0242403 and IIS0325172, and DAML program under DARPA contract F30602-00-2-0591

profiles using a recommended language (i.e. APPEL[2]), and (iii) lets P3P user agents (such as those inbuilt in a web browser) automatically verify whether a website's P3P policy conforms to user's privacy preferences.

Though the P3P framework is useful, it has not been widely adopted. Cranor et al.[3] reported that only 538 of the top 5856 websites were P3P enabled (published valid P3P policies) till May 2003. A report from Ernst & Young [6] shows that P3P adoption in the top 500 sites increased from 16% (August 2002) to 23% (January 2004). Moreover, P3P user agents are also not well adopted by users. Users are discouraged by the limited expressiveness of APPEL, and the limited number of P3P enabled websites.

We believe the key to making machine interpretable privacy policies more widely used is by improving the user-side privacy decision; we therefore propose a two-step enhancement to the P3P framework using the Semantic Web technologies and models of trust: (i) We use an RDF based policy language, Rei [8], for more effective modeling of user privacy preferences. (ii) We enhance the trust model in Web privacy. The current P3P trust model builds users' trust in websites based only on the existence of P3P policies and privacy certifiers, which are independent agencies like TRUSTe (http://www.truste.org/) who verify and certify a website's privacy practices. We argue that this trust model is insufficient and propose a new model. In our model, trust is derived not only from the conformance of a website's stated policies to a user's preferences, but also from the existence of website evaluation statements, that can be obtained by consulting trusted recommenders. In both these cases we show the utility of our ontology based approach by comparing it with their counterparts in the existing P3P framework.

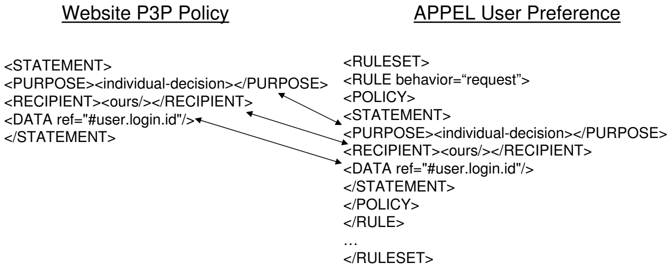
2 The Current Web Privacy Landscape

In this section we introduce the existing privacy protection mechanisms, their associated limitations and the relevance of our work to the P3P framework.

2.1 Limitations of User Privacy Preference Languages

The P3P policy is used by websites to publish information about their data collection, usage, retention and distribution policies in XML. APPEL (A Privacy Preference Exchange Language) policy is the P3P counterpart on the client side which allows users to specify their privacy preferences. APPEL is used by user agents to automatically make privacy decisions for users. The P3P vocabulary defines XML elements like RETENTION, PURPOSE etc. and their allowed values. The APPEL policy on the other hand consists of multiple RULE elements which specify user requirements for matching with P3P policies. Figure 1 shows a simple example of matching a P3P policy with an APPEL user policy, and matched elements are connected by arrows.

This approach is adopted by some P3P user agents like Privacy Bird by AT&T (http://privacybird.com) and P3P Proxy by JRC (http://p3p.jrc.it). However these user agents are rarely used by users. While one reason is the low adoption of P3P policy by websites, the other more direct reason is the



```
<STATEMENT>
<PURPOSE><individual-decision></PURPOSE>
<RECIPIENT><ours/></RECIPIENT>
<DATA ref="#user.login.id"/>
</STATEMENT>
```

```
<RULESET>
<RULE behavior="request">
<POLICY>
<STATEMENT>
<PURPOSE><individual-decision></PURPOSE>
<RECIPIENT><ours/></RECIPIENT>
<DATA ref="#user.login.id"/>
</STATEMENT>
</POLICY>
</RULE>
...
</RULESET>
```

Fig. 1. P3P-APPEL matching

inadequacy of APPEL. Agrawal et al. [1] detail the limitations with APPEL, namely its notion of logical connectives, rule ordering and matching criteria. Due to these reasons APPEL can specify only what is unacceptable and not what is acceptable for a user[1]. Their proposed solution is XPref, an XPath based language for user preferences. Though XPref solves the issues with rule matching in APPEL, it does not solve the problem of restrictive expression capabilities of APPEL. Hence we propose the use of privacy policies described in an RDF based policy language, Rei, which can make policy decisions over actions and associated restrictions modeled using a Web privacy ontology. These user policies can be evaluated over website policies published in P3P-RDFS[10] or other constraints specified by our ontology.

2.2 Limitations of the P3P Trust Model

Maintaining and building customer trust is a very important criteria for the growth of online business. A recent survey [4] by Ernst and Young suggests that 56% of online consumers believe that websites do not adhere to their privacy policies. Further, 90% of these consumers believe that an important way of increasing their trust with a website is through independent verification of privacy policies and its subsequent enforcement. Websites have resorted to different mechanisms to build and maintain this trust, like customer service, better handling of user data, text privacy policy, certification etc.

The P3P framework also attempts to build and maintain trust of consumers in websites. This is through publishing of P3P based policies and a legal entity that is accountable for specified P3P policy, namely certifiers (e.g. TRUSTe) used in human readable privacy policies. We argue that this model does not incorporate trust sufficiently. First, it is highly coupled to the presence of a privacy certifier, which is rarely used by websites. Second, in the absence of a privacy certifier the model makes implicit assumptions that the presence of P3P policies is sufficient for building trust. Other factors like website popularity, which also lead to trust in a website, are not considered. These factors are to some extent alternatives to a privacy certifier. To gather them, our approach uses a website recommender network for privacy related knowledge sharing, drawing from the popularity of similar systems like Epinions (http://www.epinions.com) and Bizrate (http://www.bizrate.com). Our system makes such a recommendation system machine understandable, by using Semantic Web languages to markup shared information.

2.3 Limitations of Web Browsers

Popular web browsers have inbuilt user agents which implement a simplification of P3P. Though Mozilla does not formalize representation of user privacy requirements, Internet Explorer provides its own language for user privacy specification. These browsers are limited only to cookie handling heuristics. We attribute the simplified approach of these browsers to the limitations of APPEL and the low P3P adoption by websites.

3 Enhancements to the P3P Framework

Figure 2 overviews the four key components of our enhancements, namely, the client using the web privacy protection framework, the website being accessed, the website recommender network and a set of components (intelligent privacy proxy, Rei Engine, XSLT transformer and the Privacy Expert) which binds all the other components together.

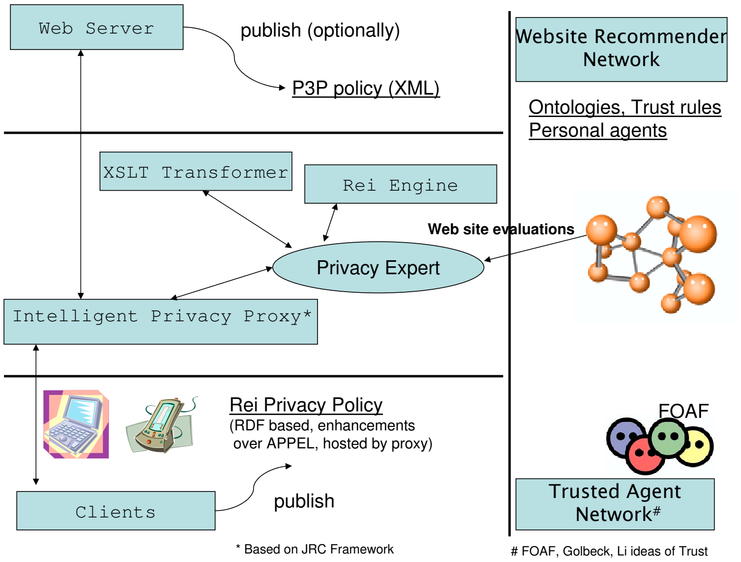


Fig. 2. The enhanced P3P Web privacy framework

3.1 Intelligent Privacy Proxy

The JRC Proxy was one of the first implementations of a P3P user agent. Users register with the proxy by publishing their user preferences in APPEL. This enables the proxy to enforce the user's preferences on all her http requests. The proxy fetches the P3P XML policy of a website and matches it against user specified APPEL policy. The entity in the proxy that does the matching is known as the APPEL evaluator. Our prototype moves the function of this evaluator to the Privacy Expert (see section 3.3). Registered users in our enhanced system are required to publish their user preferences in Rei.

3.2 Website Recommender Network

We incorporate a new trust model by using privacy recommendations/assertions provided by the Website Recommender Network. This recommender network is a social network of trusted agents that uses ontologies and rules of trust for knowledge sharing. Every user registered with the proxy has an associated Personal Social Agent(PSA) in such a network, which is responsible for gathering assertions about websites from other PSA's as well as reputation services. We term these assertions as web evaluation statements. Trust between agents on this network can be derived by using FOAF(Friend of a Friend - www.foaf-project.org) or other approaches[7, 11, 5].

3.3 Privacy Expert

The Privacy Expert (PE) is implemented as a web service allowing it to be queried by not only the modified JRC proxy, but also other privacy agents and glues the entire framework together. The main function of the PE is to make privacy decisions of the user. It takes the following inputs, namely, P3P policy, the registered username and her Rei user preference, and the website to be accessed. Then the PE collaborates with other web services to make a privacy decision. For example it uses the XSLT transformer web service[12] to convert P3P from XML to RDF. It can also query the corresponding PSA for website evaluation statements. The privacy decision is made by the Rei policy engine web service.

3.4 Remarks

A notable point in the entire framework is that no changes are required from web servers, making our scheme backward compatible, as it were. We recognize that describing privacy policies expressed in Rei/RDFS is not something that an average user will be able to do. However, there are ongoing efforts by researchers to either learn user preferences from observing their behavior (web mining on the client side), or at least provide graphical interfaces and templates for policy specification.

4 User Privacy Preference Specification

In this section we introduce Rei and detail its usage for user privacy preference specification. We also compare it with APPEL and Xpref.

4.1 Rei Policy Language

Rei is a declarative policy language, represented in RDFS (recent versions support OWL), which includes notions of logic like variables for describing different kinds of conditions. It is modeled on deontic concepts of rights, prohibitions, obligations and dispensations [8]. We identify the features that make Rei a powerful language in the Web privacy domain as the following:

- **Ontological Modeling.** Rei policies are based on ontologies that model users' privacy preferences. The policy language has domain independent ontologies but can also reason over specific domain dependent ontologies. This ontology based approach provides rich semantics for specification of highly expressive policies.
- **Scope for future extensions.** Rei allows rule specification involving obligation and delegation management. Obligations are promises made by a website (e.g. e-mail notification on privacy policy updates) and delegations (e.g. "we share information with our trusted partners who do not have an independent right to further share this data") are policies concerning the distribution of data. Since websites cannot publish such information using P3P, we do not detail them in this paper.
- **Rule engine.** Associated with Rei is a rule engine that interprets and reasons over the policies, related speech acts and domain information expressed in RDFS to make decisions about applicable rights, prohibitions, obligations and dispensations.

4.2 Rei and User Preference Specification

Before detailing our Rei-based approach, we compare its language features with the existing approaches – APPEL and Xpref, and summarize them in table 1.

Table 1. A language feature comparison of APPEL, Rei and XPref

	APPEL	**Rei**	**Xpref**
RootElement	RULELIST	Policy	RULELIST
RuleElement	RULE	Granting	RULE
Actions	request, block, limited	arbitrary domain action	request, block, limited
Ontologies	Not supported	RDFS	Not supported
Constraints	P3P Specific	Domain ontology	P3P Specific
Rule Priority	Not supported	specified by RulePriority	Handled using "XPATH"

Rei has specific *domain independent ontologies* that mandate the policy syntax which is represented in RDFS(http://www.cs.umbc.edu/~lkagal1/rei/ontologies). Figure 3 shows a subset of classes in Rei ontologies that are sufficient for specifying user policies in the Web privacy domain. It uses the following notations: classes are depicted by oval nodes; properties are depicted by directed edges from the domain to range of the property; dashed edges associate the class *Constraint* to different possible types of constraints; all entities with a white background have a counterpart in APPEL; and shaded entities are enhancements provided by Rei.

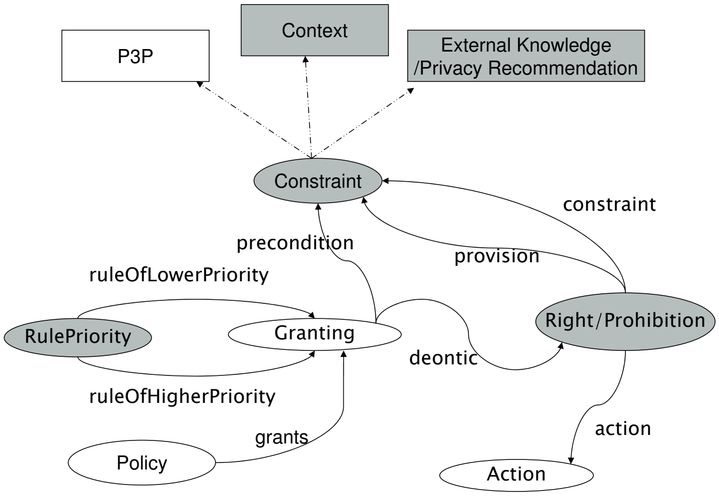


Fig. 3. Rei concepts used in Web Privacy

Policy. An instance of *Policy* is counterpart to the APPEL *RULELIST* element and is used to describe a user's privacy preferences.

Granting. An instance of *Granting* is the counterpart to the APPEL *RULE* and represents an individual rule of a user's privacy preferences.

Action Each policy rule is specified on certain actions using the *deontic* property. *Action* can be one of *Right, Prohibition, Obligation or Dispensation* and is defined based on domain specific ontologies. For example, *Actions* as they relate to Web privacy can be categorized - request(allow), block, limited, request-prompt, limited-prompt. This provides a direct mapping to actions in APPEL. Other actions can also be specified based on the capabilities of the enforcement mechanism (proxy server, web browser etc.)

precondition, provision, constraint. Rei policies decide allowing certain actions (i.e. user sharing information when accessing a website) based on constraints specified by the *precondition*, *provision* and *constraint* properties. *Preconditions* for a rule (e.g. web resource being accessed is not an activeX control) allows filtering of rules before checking for constraints. Rei also provides the ability to specify *provision* which can be used to specify obligations for a particular *Action*. One such example is the obligation of a web browser (enforcement mechanism) that cookies should be deleted after the current session. All the other constraints are defined by the *constraint* property. All these properties in the range of *Constraint*

Constraint. Rei can specify a wide range of constraints through the inclusion of domain specific ontologies. A constraint is of the form "x is a type of website" or more generally "x has an attribute y with value z". As shown in figure 3, APPEL and Xpref let a user specify constraints only on P3P policies published by websites. Rei allows constraints on other domain specific knowledge like context information (e.g. IP address of the client) and privacy related statements (e.g. popularity of the site) from the web evaluation ontology.*Constraints* can be either simple conditions acting on a single variable or complex constraints using a combination of simple conditions. Constraints can be grouped together using *AndConstraint*, *OrConstraint* and *NotConstraint*. Domain specific ontologies over which policies are specified like user context, website, appel-actions etc. are available at http://www.cs.umbc.edu/˜kolari1/ontologies/.

RulePriority. Two properties *ruleOfHigherPriority* and *ruleOfLowerPriority* associate the instance of *RulePriority* to instances of *Granting*. To specify priorities between more than two instances of *Granting*, multiple instances of *RulePriority* can be used and cascaded. Rei also provides modality preference i.e. positive over negative or vice versa which can be used to specify that rights(e.g. allow access) have priority over prohibition(e.g. block access) or vice versa.

Logical connectives. Since Rei allows the use of operators *and, not* and *or* and uses Prolog as the rule engine, it can represent all kinds of logical connectives and unambiguously reason over them.

5 Enhancements to the P3P Trust Model

In order to solve the limitation of the current P3P trust model, we consider factors that lead a user to trust a website, and formalize these factors through the Web Evaluation Ontology. The ability of our RDF based user preference language to make use of domain specific information for privacy decisions, over and above the published P3P policies, allows easy integration of this trust model into the P3P framework. We introduce a mechanism for instantiation of this ontology using a trusted recommender system for website evaluation statements. Such a system also allows verification of the instances through consensus.

5.1 Website Evaluation Ontology

Trusting a website's privacy practice includes two aspects: trusting if the website's privacy policy satisfies the user privacy preference, and trusting if the website adheres to its own privacy policy. A trust judgment requires statements characterizing websites: either the statements in the website's P3P policy, or the *website evaluation statements* gathered from external sources. A website evaluation statement provides evaluation of a website's privacy policies and practices and its reputation. This allows a user to make trust judgments and hence privacy decisions, even in the absence of P3P policy. For example, most of the ".edu" websites have an acceptable privacy policy. We have developed an ontology (http://www.cs.umbc.edu/~kolari1/ontologies/Website.rdfs) that models two categories of website evaluation statements - the meta privacy evaluation statements and the implicit privacy evaluation statements.

The main focus of the meta privacy evaluation statements is building trust in a website's privacy practices itself. We identify the following properties as being in this category.

- hasPrivacyPolicy - is the URI of the website's human readable privacy policy. The presence of text privacy policy has to be incorporated into automatic decision making. This information can be obtained by a web crawler.
- hasP3PPolicy - is the URI of the website's P3P policy. This information shows the website's commitment to automatic privacy protection of users.
- privacyPolicyCertifiedBy - links a website to the URI of its privacy certifier. If a website publishes P3P, this information can be obtained directly, otherwise scraping of the the text policy page is required.
- privacyPolicyEnforcedBy - is the URI for the internal system that enforces a website's privacy policy. However websites might not be willing to share such information.

The main focus of implicit privacy evaluation statements is building trust in a website itself, which to some extent implies trust in the website's privacy practices. We identify the following properties to be in this category.

- domainSuffix - is the suffix of a website's domain name, such as ".com", ".gov", and ".edu". For example an educational website with domainSuffix .edu, would rarely set cookies and use them in ways that might breach user privacy.
- owner - is the URI of a person or an organization owning the website. A website owned by a highly reputed company (such as a well-known bank) might be trusted to have good privacy protection mechanisms.
- reputation - represents the overall rating of a website offered by online reputation services, such as Google PageRank, Bizrate rating, and Epinion rating. They compute ratings differently. For example, Google uses the PageRank algorithm, Epinion aggregates individual user ratings and a PSA relies on recommenders.
- popularity - refers to the number of users who review the website with rating. Intuitively, it shows the confidence about the reputation of a website. Link popularity can be obtained by web scraping sites like http://www.google.com, http://www.linkpopularity.com, http://www.popdex.com etc. which gives the number of inlinks to a particular page. On the same lines the Bizrate rating uses the number of reviewers rating a website.

– subDomainOf - gives information about the website to which this site is affiliated to. For example images.yahoo.com is affiliated with www.yahoo.com, a highly trusted site.
– isBasedOutOf - links a website to a URI of a geographical location in which the host machine of the website is located. This location information can be obtained from online services that maps IP address to spatial location. Different countries have different social value systems; hence they view user privacy protection in different ways.
– lawAccountability gives information about the privacy laws which are in effect for a particular website. This could be because of a particular website being hosted at a particular country or state. A user might be confident of privacy laws in U.S and not very certain about the one's in Asia.
– policySimilarTo - specifies the similarity between two sites as it relates to their privacy practices. For example www.slashdot.org specifies that its privacy policies are based on that of its parent company, namely the OSDN network. A user trusting OSDN's privacy policy will in turn trust that of slashdot's.
– domainOfService - represents website classification based on the type of service provided like informational site, advertising site, search site etc. This information can be obtained from online directory servers to make privacy decisions and provides higher granularity over the domainSuffix introduced earlier. The kind of service a website offers can give valuable information about data that has to be protected and released. For example contact and clickstream information can be provided to a book seller (if the user is a book enthusiast), whereas for an advertising site, (e.g. doubleclick) the release of clickstream information should be controlled as this might lead to user tracking across multiple websites.

5.2 Trust Based Recommender System

Our trust based recommender system is analogous to online recommendation systems. Though existing online review/rating systems are geared towards customer satisfaction in online shopping domain, we believe that in the future privacy will also be an important issue in online communities. An important feature of our recommender system is that all agents are implemented as (semantic) web services, which is the trend of future web communication. As shown in figure 4, we are currently implementing a trust based recommender system that allows users to exchange their privacy evaluation statements based on their own experiences as well as obtain recommendations from online reputation services. Our recommender system consists of two categories of agents: *web information sources* and *personal social agents.*

Web information sources collect information from the Web, and encode information using the Semantic Web technologies. Currently, we envision three types of web information sources. (i) *Reputation systems* (such as Epinions, Bizrate, and eBay.com) collect customer satisfaction reports and generate overall rating about commercial websites[1]. (ii) *Trusted third parties* provide "objective"

[1] We have implemented information extraction agents for Bizrate.com

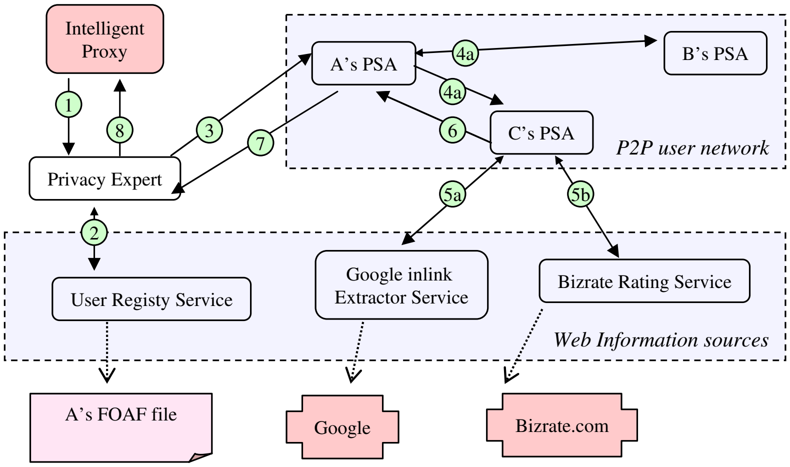


Fig. 4. The trust based recommender system

evaluation about websites, which is trusted by most online users. For example Google reports statistics about the inlinks to a web page based on a significant amount of observations, and TRUSTe.org provides privacy certification based on its privacy verification mechanism. (iii) *User registry service* can automatically discover users' personal web services and their trusted peers(web services) by analyzing their online FOAF profiles.

Personal social agent (PSA) interfaces a user with the online community. It allows a user to gather knowledge from the other PSAs and web information sources. By maintaining and evolving trust relations between agents, the PSAs can form a peer to peer (P2P) social network in a distributed environment. PSA discovers the other users' PSA address through a user registry service. We assume trust can be derived between any pair of PSAs through trust network inference [11,7,13,5].

An example walkthrough *The message flow in figure 4 depicts how the trust based recommender system helps the intelligent proxy make privacy decision. When the intelligent proxy needs to make a privacy decision for a particular user, it sends message 1 to a PE. The PE then queries "user registry service" for user A's PSA's address with A's name in message 2. When the PE obtains the address successfully, it forwards the query to A's PSA in message 3. A's PSA consults two peers B and C in message 4a and 4b (assume A's PSA already knows B and C's PSAs' addresses, these could be specified in FOAF profiles). C further consults two web information sources for website rating and inlink statistics in message 5a and 5b. Then C's PSA passes its web evaluation statements and sends it to A's PSA in message 6. Finally, A's PSA combines the statements from B* *and C and passes the result back to the PE in message 7. The PE makes a privacy decision by querying the Rei engine and passes this result to the intelligent proxy in message 8.*

5.3 Uncertainty of Constraints

Since human users prefer simple Boolean inference, the "Constraint" in a Rei policy takes Boolean valued statement as input. However, the privacy evaluation statements are inherently uncertain (neither absolutely true nor absolutely false) because they are subjective judgments and propagated by trust. Here, we interpret *trust in statement* as the possibility that a statement is true, and *trust in information source* as the possibility that an information source is truthful (i.e., it always tells true statements). An agent could reach better inference result if it were to exploit this uncertainty.

In order to manage the uncertain statements at software agent level and then pass the Boolean assertions to Rei, we currently adopt a simple approach based on Zadeh's fuzzy logic. Other possible approaches for handling uncertainty include possibility theory, probability theory, and etc. Basically, fuzzy membership functions are used to describe the degree of trust in a statement. Fuzzy conjunction (min) and disjunction (max) are used to derive the degree of trust in "and-constraint" and "or-constraint" in Rei respectively. The degree can be used to order rule priority or resolve conflict. (e.g. given a set of conflict statements, we can simply select the one with the largest degree.). We use threshold based defuzzification to determine the Boolean value of statements.

6 A Use Case for the Enhanced P3P Framework

Cookies set on client machines are very useful for online vendors, from being enablers of online shopping to tracking of user browsing behavior. Though cookies are browser dependent, they are independent of internet service provider and client location. As an implication of this, a notebook user's activities can be tracked across locations based on her IP address and past browsing preferences. Commonly accessed websites like e-mail and advertising can reconstruct the entire travel history of a particular user. Further an e-mail service like GMail[2] can associate this information with the e-mails being read by the user to identify the purpose of the visit.

An option in the above mentioned scenario would be to use cookie cutters to shield the user from such implicit data collection. However this requires the user to manually deny cookies to certain websites. To automate control so that a user can decide when cookies can be released to websites, we list the criteria (constraints) of a typical user as follows. Similar constraints can be specified for other implicit and explicit data collected.
(i) Trustable P3P constraint - The website publishes a P3P policy and specifies that data collected is user IP and click-stream, its usage is website tailoring, and retention period of this data is no-retention(deleted after current session) and

[2] www.gmail.com

the P3P policy is certified by Trust-E. If it is not certified, the overall rating from the web evaluation ontology is 5 (assuming that rating has integral value and is between 0 and 5). This allows trusting a website through either a certifier or a web evaluation statement.
(ii) Trusted website constraint - The website is one of a set of websites which is highy trusted by the user, be it her bank, the company she works for or her home page. For a bank or the employer, cookies might be an important way of accountability should there be a conflict in transactions.
(iii) Trusted domain constraint - The website is either of type(domainSuffix) "edu" or" gov" and is based out of USA. The user is willing to share information as she is confident of privacy practices of such websites.
(iv) DomainOfService Negative constraint - The website is neither of type(domainOfService) portlet(e.g. Yahoo, AOL) nor of type advertising. Such websites if allowed direct access might create and store a history of users travel habits.
(v) DomainOfService constraint - The domainOfService of the website is "travel". A travel related site could give her hints about her travel plans and save her both time and money. When the user is in a new city, allowing cookies (user preference) to a site providing local information or any other travel related site might be useful for the user, which in this case might present a user a free ticket to a concert when in Baltimore. Further, information that the website caters to a particular location for example Maryland, can also be used if available, so that cookies are allowed to only such websites. This will allow specification of more specific and stricter constraints. All of the above mentioned constraints are based on the assumptions that - user privacy is being protected when the user is travelling i.e context information browsingFrom has value "away", cookies are not exchanged with a site during the process of querying for P3P policy itself and the website is not one where a user is required to login to access provided services(e.g. Gmail).
The ability of our user preference language to incorporate external knowledge, which in the above scenario is context knowledge and trust evaluation statements lets a user specify preferences of the above mentioned complexity with ease. Figure 5 depicts the actual policy specification of the required rules using Rei. One of the rule incorporates all the constraints of a user and another rule is the default. In the case of typical user preferences many such rules are specified with rule priorites deciding the order of rule matching.

WebPolicy is an instance of *Policy* and specifies two rules for actions related to Web Privacy, namely *Block* and *Request*. *GrantingDefault* specifies the default rule, and *GrantingRight* specifies the rule for the constraints governing *Request* action. *RulePriority* sets higher priority to the *GrantingRight* rule. The default rule has a lower priority and blocks access to a website. Note that in this example, there are preconditions that the resource being accessed is a html page and the user's context is "away". This shows how we can use different policy rules when acessing web services [3], images, active-X controls or just when browsing from home.

The action associated with the deontic concept *Right* is *Request* i.e. allow access to a website. The *Right* has constraints. *Constraints* are underlined, to point

[3] P3P for web services is a currently evolving specification – http://www.w3c.org/

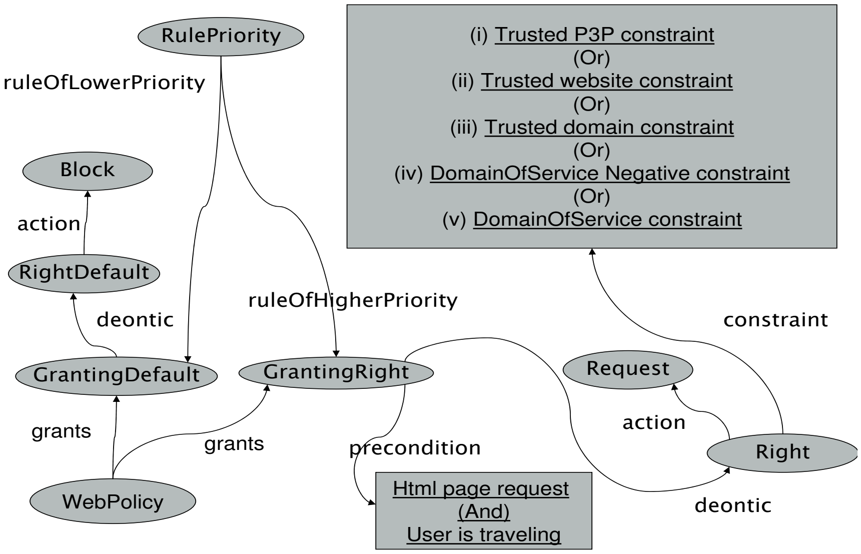


Fig. 5. A Rei rule using trust and context

to the fact that they are represented by English-like statements for clarity. In reality they are asserted as triples into our knowledge base, based on the domain ontology. The specified constraints are the one's listed above as a typical user preferece in such a scenario. All constraints are specified in rectangular blocks to clearly delineate them from other parts of the user preference. Constraints are grouped together using *BooleanConstraint* provided by Rei. The complete example is available at
http://www.cs.umbc.edu/˜kolari1/ontologies/examples/comprehensive.rdf. Other examples which depict user preference examples for low, high and medium level privacy protection along with their descriptions are available at
http://www.cs.umbc.edu/˜kolari1/ontologies/examples/.

Given such a Rei policy, the Rei engine makes decisions based on satisfying constraints. Since *GrantingRight* has higher priority, constraints specific to it are first checked. If these constraints fail, the action *Block* specified by *GrantingDefault* is fired, which specifies the action "block". The action "block" could be interpreted in different ways by the client software. It could mean denying access to site, blocking cookies, or requiring obligations from browsers or websites that cookies and information that they provide are deleted after every session. Enforcing a particular action depends on the capability of the browser and websites. The JRC proxy denies access to a website.

7 Conclusion

The ultimate goal of Web privacy protection is to safeguard a user's personal information automatically. Towards this end, our enhancements to the P3P framework make the following contributions. In comparison with APPEL, we show that the RDF based user privacy preference specification is more expressive and suitable in the Web privacy domain. In order to make the P3P trust model effective, we introduce a *website evaluation ontology* and adopt a trust based recommender system for propagating web evaluation statements. We show how our system is effective even in the absence of published P3P by websites. This will encourage adoption of P3P user agents by users and in turn result in better adoption of P3P policies by websites.

The overall system also shows the effectiveness of the Semantic Web concepts in the domain of privacy protection. The primary motivation of the W3C P3P project is to automate the client side privacy decision. For such a framework to be widely adopted, machine understandable information from multiple sources have to be aggregated and used. The widespread deployment of the Semantic Web will make such privacy decisions easier, without the need of implementing web scraping services on a per-site basis.

References

1. R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu. An xpath-based preference language for p3p. In *Proceedings of the twelfth international conference on World Wide Web*, pages 629–639. ACM Press, 2003.
2. L. Cranor, M. Langheinrich, and M. Marchiori. A p3p preference exchange language 1.0. W3C Working Draft 15 April 2002.
3. L. F. Cranor, S. Byers, and D. Kormann. An analysis of p3p deployment on commercial, government, and children's web sites as of may 2003. Technical report, AT &T Labs-Research, 2003. prepared for the 14 May 2003 Federal Trade Commission Workshop on Technologies for Protecting Personal Information.
4. J. R. DeVault, D. Roque, jay Rosenblum, and K. Valente. Privacy promises are not enough. Technical report, Ernst&Young, 2001.
5. L. Ding, L. Zhou, and T. Finin. Trust based knowledge outsourcing for semantic web agents. In *Proceedings of IEEE/WIC International Conference on Web Intelligence*, 2003.
6. Ernst&Young. P3p dashboard report: Top 500 p3p dashboard, 2004.
7. J. Golbeck, B. Parsia, and J. Hendler. Trust networks on the semantic web. In *Proceedings of Cooperative Intelligent Agents*, 2003.
8. L. Kagal, T. Finin, and A. Joshi. A policy language for a pervasive computing environment. In *Proceedings of IEEE 4th International Workshop on Policies for Distributed Systems and Networks*, June 2003.
9. H. Kargupta and P. Chan, editors. *Advances in Distributed and Parallel Knowledge Discovery*. MIT/AAAI Press, 2000.
10. B. McBride, R. Wenning, and L. Cranor. An rdf schema for p3p. W3C Note 25 January 2002.
11. M. Richardson, R. Agrawal, and P. Domingos. Trust management for the semantic web. In *Proceedings of the Second International Semantic Web Conference*, 2003.
12. R. Wenning. (w3c privacy activity lead) xslt for p3p rdf. Personal Communication, March 10, 2004.
13. B. Yu, M. Venkatraman, and M. P. Singh. An adaptive social network for information access: Theoretical and experimental results. *Journal of the Applied Artificial Intelligence*, 17(1), 2003.