A Privacy Policy Framework – A position paper for the W3C Workshop of Privacy Policy Negotiation

Paul Madsen, NTT

Marco Cassasa Mont , HP Labs

Robin Wilton, Sun Microsystems

Overview

Dictionary.com defines 'policy' as:

> *A plan or course of action, as of a government, political party, or business, intended to influence and determine decisions, actions, and other matters.*

In the context of information security, 'privacy' is often defined as:

> *The interest an individual has to control how information about them is collected, used, and shared.*

Consequently, we define here a 'privacy policy' as:

> *A plan or a course of action intended to influence and determine decisions, actions, and other matters concerning the collection, use, and sharing of an individual's personal information.*

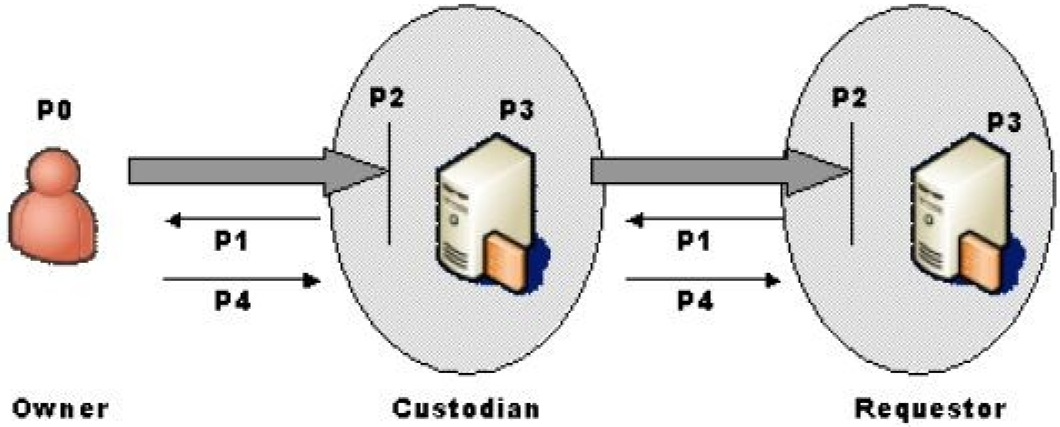
A privacy policy infrastructure will provide to individuals the necessary control over their information, as well as allow the business collecting, using, and potentially sharing that information, to ensure that they abide by appropriate regulatory requirements regarding such data.

This paper discusses the different aspects of such a privacy policy infrastructure, proposes a taxonomy for these aspects, and explores the applicability of various XML-based policy languages to these different aspects.

We conclude with a more detailed description of how Liberty's ID-WSF specifications can be applied to the exchange of privacy policy information between principals (specifically, the data subject, the data custodian and the data requester).

Privacy Policy Taxonomy

We can identify different aspects of privacy policy. The diagram below illustrates.



The different types of privacy policies labeled above are described here:

P0 - policy preference (e.g. If I provide my credit card number I expect ...)
P1 - policy promise (e.g. I will use your credit card number for ...)
P2 - governs acceptance (e.g. Don't accept credit card number unless)
P3 - governs internal use & release (e.g. Only share credit card number when ...)
P4 - governs subsequent use/release (e.g. If you receive the credit card number, you must delete after 3 days)

The arrows in the diagram represent the flow of PI data and privacy policies. PI data is shown as flowing from the owner to the custodian and then, potentially, to subsequent external applications in a separate policy domain. For the different policies, P0, P2, and P3 are 'local', i.e. they are not communicated between actors. P1 and P4 are communicated between entities.

Policy Syntax

There are a number of XML-based syntaxes that may be relevant for capturing privacy policies. This sections explores the applicability of some of these languages to the 5 different types of privacy policy presented above.

P3P

Platform for Privacy Preferences (P3P) is a standardized set of multiple-choice questions, covering all the major aspects of a Web site's privacy policies. Taken together, they present a clear snapshot of how a site handles personal information about its users. P3P-enabled Web sites make this information available in a standard, machine-readable format. P3P enabled browsers can "read" this snapshot automatically and compare it to the consumer's own set of privacy preferences. P3P enhances user control by putting privacy policies where users can find them, in a form users can understand, and, most importantly, enables users to act on what they see.

XACML

Extensible Access Control Markup Language (XACML) is an OASIS standard that describes both a policy language and an access control decision request/response language (both written in XML). The policy language is used to describe general access control requirements, and has standard extension points for defining new functions, data types, combining logic, etc. The request/response language lets you form a query to ask whether or not a given action should be allowed, and interpret the result. The response always includes an answer about whether the request should be allowed using one of four values: Permit, Deny, Indeterminate (an error occurred or some required value was missing, so a decision cannot be made) or Not Applicable (the request can't be answered by this service).

An XACML policy has the following components:

- Rule - targeted atomic policy element that serves as container for condition and effect.
- Conditions - a boolean function over a combination of subject, resource and environment attributes.
- Effect - identifies the intended consequence if the rule's condition evaluates to 'true'.
- Target - identifies the set of decision requests (subject, action, and resources) that the rule/policy is intended to evaluate.
- Policy - one or more rule(s), an algorithm for combining them, and optionally a target and obligations.
- Obligations - actions that must be performed by the requesting PEP on either a permit or deny response.

XACML privacy Profile

The "Privacy policy profile of XACML v2.0" was approved as a full OASIS Standard on 1 February 2005. A copy of the profile is available at: http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-privacy_profile-spec-os.pdf

The profile defines two standard attributes

urn:oasis:names:tc:xacml:2.0:resource:purpose

This attribute indicates the purpose for which the data resource was collected. The attribute value MAY be a regular expression.

urn:oasis:names:tc:xacml:2.0:action:purpose

This attribute indicates the purpose for which access to the data resource is requested.

This is in effect a generic privacy rule in that it specifies that the stated purpose for which access to a resource is being requested must match the purpose associated with that resource - this rule independent of what the specific purpose (e.g. marketing) or what the resource may be. It is significant in that these are also the terms in which, for example, European Data Protection laws tend to be framed; that is, that PII collected for one purpose may not be re-used for other purposes.

EPAL

The Enterprise Privacy Authorization Language (EPAL) is an interoperability language for exchanging privacy policy in a structured format between applications or enterprises.

EPAL is designed to make it easier for enterprises to translate their privacy policies into machine-readable descriptions of data handling procedures. For instance, EPAL lets developers express a natural language statement such as "Members of the physician group can read protected health information for the purpose of medical treatment, only if the physician is the primary care physician and the patient or the patient's family is notified in advance" in a language that applications and privacy management tools can understand.

A EPAL policy document consists of three main sections:

- Policy Information: This is used to identify the policy. It consists of information such as Issuer, Version Number, Start Date, End Date, Replacement Policy Name, Replacement Policy Version.
- Definitions: This defines all of the possible components that can be used in the following rules. Here is where Data Users, Data Categories, Purposes, Actions, Context Models, Conditions and Obligations are defined.
- ALLOW or DENY: Rules to define whether Data Users are ALLOWed or DENYed to perform Action on Data Category for Purpose under Conditions.

ODRL

The Open Digital Rights Language (ODRL) is a "vocabulary for the expression of terms and conditions over digital content including permissions, constraints, obligations, conditions,offers and agreements with rights holders."

The ODRL specification supports an extensible language and vocabulary (data dictionary) for the expression of terms and conditions over any content including permissions, constraints, requirements, conditions, and offers and agreements with rights holders.

All ODRL specifications are available without any obligations and have no licensing requirements. The latest version of the ODRL specification (Version 1.1) has been co-published by W3C as a W3C NOTE. Additionally, ODRL has been officially accepted by the Open Mobile Alliance (OMA) as the standard rights expression language for all mobile content.

Applicability

The following table presents an assessment of the applicability of the policy expression languages to the 5 different types of privacy policies discussed earlier.

Note: a 'no' within a column should not be construed as meaning that the corresponding policy syntax is incapable of supporting that policy aspect, rather that it may not be optimally designed for that aspect.

	P0	***P1***	***P2***	***P3***	***P4***
P3P	No	Yes	No	No	No
XACML	No	No	Yes	Yes	Yes
EPAL	No	No	No	Yes	No
ODRL	No	No	No	No	Yes

Liberty ID-WSF as a Privacy Policy Framework

Liberty ID-WSF defines protocols by which attributes can be requested and subsequently shared, ID-WSF focuses on the 'on-the wire' aspects of privacy policy, specifically P1 and P4 type policies. While ID-WSF assumes that the other types of privacy policy (specifically P3) are in place, it defines no specific mechanisms in their support.

UsageDirectives SOAP Header block

Liberty's SOAP Binding defines the <UsageDirectives> element to carry P1 and P4 policies.

Message senders may add one or more <UsageDirective> header blocks to the SOAP Header of the message being sent.

A <UsageDirective> appearing in a request message expresses intended usage. A <UsageDirective> appearing in a response expresses how the receiver of the response is to use the response data. A <UsageDirective> in a response message containing no ID-WSF response message data, a fault response for example, may be used to express policies acceptable to the responder.

The <UsageDirective> element acts as a container for P1 and P4 policy statements. Liberty did not specify what form such statements within the <UsageDirective> element might take , ie Liberty neither defined its own policy language nor normatively referenced some other standard.

An example of the use of the <UsageDirective> element is shown below. Here a request for some principal's address is supplemented by a policy statement indicating that the address will be used in a manner consistent with the constraints of the European Union's privacy regulations.

```
<S:Envelope xmlns: s="http://schemas.xmlsoap.or g/soap/envelope/"
xmlns:sb="urn:liberty:wsf:soap-bind:1.0"
xmlns:pp="rn:liberty:idpp:1.0">
<S:Header>
<UsageDirective
         id="directive1000"
         ref="#datarequest001"
         S:mustUnderstand="1">
         <cot:PrivacyPolicyReference xmlns:cot="http://circle-of-trust.com/isf">
         http://circle-of-trust.com/policies/eu-compliant
         </cot:PrivacyPolicyReference>
</UsageDirective>
</S:Header>
<S:Body>
<pp:Query id="datarequest001" xmlns="urn:liberty:pp:1.0">
<pp:Resource>data:d8ddw6dd7m28v628</pp:Resource>
<pp:QueryItem>
<pp:Select>/pp:PP/pp:AddressCard</pp:Select>
</pp:QueryItem>
</pp:Query>
</S:Body>
</S:Envelope>
```

If this level of policy detail was sufficient and it met the P3 policy requirements of the provider hosting the address information, the address information would be released. If not, the provider would return a SOAP fault indicating failure, and potentially also return policy identifiers under which the information would be released.

Provisional conclusions

At this stage, our analysis of the range of existing PPELs is a comparatively high one and does not support a conclusive recommendation of any of them over the others - though we feel that the taxonomy presented does offer a basis for systematic analysis in more detail.

Market experience suggests that, where PPELs are deployed at all currently, most implementations are in the "P3" area – that is, definition and enforcement within the Zone of Control (ZoC) of a single server or organisational domain. The trend towards web services and collaborative service provision increases the need for privacy preferences to extend beyond that boundary, and into the cross-organisational realm – in other words, into the areas of "P1" and "P4".

There are also those who argue, on grounds of the inchoate concept of 'user centricity', that P0 should be high on the list of 'next steps'.

In our view, while a fully 'end to end' implementation from P0 to P4 is still some way from technical viability, a scheme which encompasses P3 (based on XACML), with P1 and P4 (based on XACML payloads over ID-WSF messaging) is entirely acheivable and would represent a worthwhile evolution of current capabilities.

References

[LawID] eds. (2003). "LawID Position Paper - Workshop on the long-term future of P3P ," http://www.w3.org/2003/p3p-ws/pp/lawids.html,

[MSFT] eds. (2003). "Microsoft Position Paper - Workshop on the long-term future of P3P ," http://www.w3.org/2003/p3p-ws/pp/microsoft.html,

[PRIME] eds. (2003). "Privacy and Identity Management for Europe," http://www.prime-project.eu.org/,

[W3C] eds. (2003). "Enterprise Privacy Authorization Language (EPAL 1.2) W3C Member Submission 10 November 2003," http://www.w3.org/Submission/2003/SUBM-EPAL-20031110/,

[IBM] eds. (2003). "IBM Research Position Paper - Workshop on the long-term future of P3P - Translating EPAL to P3P - How to keep enterprise privacy promises in sync with the actual practic," http://www.w3.org/2003/p3pws/pp/ibm2.html,

[CLARK] eds. (2003). "Introduction to Dataveillance and Information Privacy, and Definitions of Terms ," http://www.anu.edu.au/people/Roger.Clarke/DV/Intro.html,

[EPAL] eds. (2003). "IBM Research Position Paper - Workshop on the long-term future of P3P - Enterprise Privacy Authorization Language (EPAL) - How to Enforce Privacy throughout an Enterprise," http://www.w3.org/2003/p3pws/pp/ibm3.html,

[HPLABS] eds. (2003). "HP Labs Position Paper - Workshop on the long-term future of P3P - On the Importance of Accountability and Enforceability of Enterprise Privacy Languages," http://www.w3.org/2003/p3p-ws/pp/hp1.pdf,

[SOAPBINDING] eds. (2003). "Liberty ID-WSF SOAP Binding Specification," http://www.projectliberty.org/specs,