Privacy Enforcement Architectures for an e-Business Environment

By

Dianshu Hu

A thesis submitted to

the Faculty of Graduate Studies and Research

in partial fulfillment of

the requirements for the degree of

Master of Computer Science

Under the auspices of

the Ottawa-Carleton Institute for Computer Science

School of Computer Science

Carleton University

Ottawa, Ontario, Canada

March 2011

Your file Votre référence
ISBN: 978-0-494-81655-4
Our file Notre référence
ISBN: 978-0-494-81655-4

Abstract

Privacy is major concern for both web users and e-businesses. Research and implementation have begun in the areas of privacy enforcement architecture like the Architecture for Privacy Enforcement using XML (APEX) and the complimentary mechanisms of APEX. This thesis provides the design of the Enhanced Architecture for Privacy Enforcement using XML (EAPEX) which takes into consideration user privacy preferences to enforce the negotiated privacy practices pertaining to web users' private data and to address web users' privacy concerns in an e-business environment. This thesis research also develops the design of new system architecture, called the Architecture for Privacy Enforcement using Policy Based Encryption (APEP), for negotiated privacy practices enforcement in the e-commerce environment using on a promising Policy Based Encryption (PBE) system. A major benefit of this architecture design is that it seems to have less system complexity because privacy control is automatically achieved by using the PBE.

Acknowledgements

I wish to sincerely thank Dr. Carlisle Adams for offering me this interesting thesis topic, for providing his insights and suggestions that guided me in my thesis research and for providing his dedication, time and encouragement that carried me through the thesis project.

I am also heartily thankful to my supervisor, Dr. Evangelos Kranakis. Without his encouragement, guidance and support from the initial to the final level, this thesis would not have been possible.

Table of Contents

List of Figures

List of Tables

List of Acronyms

APEP	Architecture for Privacy Enforcement using Policy Based Encryption
APEX	Architecture for Privacy Enforcement using XML
APPEL	A P3P Preference Exchange Language
APPT	Automated Privacy Policy Transformation mechanism
ATPX	Automated Transformation from P3P to XACML
ATXP	Automated Transformation from XACML to P3P
CDNF	Conjunctive-Disjunctive Normal Form
CNF	Conjunctive Normal Form
CSR	Customer Service Representative
DAO	Data Access Object
DE	Decision Engine
DNF	Disjunctive Normal Form
DOM	Document Object Model
EAPEX	Enhanced Architecture for Privacy Enforcement using XML
EPAL	Enterprise Privacy Authorization Language
EUPI	Encrypted User Private Information
GUI	Graphical User Interface
I&AM	Identity and Access Management system
IBE	Identifier Based Encryption
ID	Identity
IOPEE	Inter-Organizational Privacy Enforcement Engine
IOPP	Inter-Organizational Privacy Policy
IP	Internet Protocol
JDBC	Java DataBase Connectivity
JEE	Java Enterprise Edition platform
LPEE	Local Privacy Enforcement Engine
LPP	Local Privacy Policy
NRC	National Research Council
OASIS	the Organization for the Advancement of Structured Information Standards
OPCP	Organizational Privacy Control Policy
OPEE	Organizational Privacy Enforcement Engine
OPP	Organizational Privacy Policy
P3P	Platform for Privacy Preferences
P3PPEE	Privacy Enforcement Extension for P3P 1.1
PAP	Policy Administration Point
PBE	Policy Based Encryption
PCP	Privacy Control Policy
PDP	Policy Decision Point

PEA	Privacy Enforcement Architecture
PEE	Privacy Enforcement Engine
PEM	Privacy Enforcement Model
PEP	Policy Enforcement Point
PII	Personally Identifiable Information
PIP	Policy Information Point
PIPEDA	Personal Information Protection and Electronic Document Act
PPA	Privacy Policy Architecture
PPESeal	Privacy Protection Enforcement Seal
PPG	Privacy Policy Group
PTDE	Privacy Transformation and Decision Engine
RSA	the public-key encryption algorithm by Rivest, Shamir and Adleman
SAML	Security Assertion Markup Language
SOAP	Simple Object Access Protocol
SSL	Secure Sockets Layer
TE	Transformation Engine
TTP	Trusted Third Party
UPCP	User Privacy Control Policy
UPI	User Private Information
UPIG	User Private Information Group
UPP	User Privacy Preference
W3C	World Wide Web Consortium
WPEE	Web Privacy Enforcement Engine
WPP	Web Privacy Policy
XACML	eXtensible Access Control Markup Language
XML	Extensible Markup Language
XPCOM	Cross-Platform Component Object Model
XSLT	eXtensible Stylesheet Language Transformation
XUL	XML-based User-interface Language

Chapter 1 Introduction

Many businesses are moving away from paper-based customer records to use electronic customer profile records that can be directly processed by specialized software thus greatly enhancing data processing efficiency. Using electronic records also facilitates aggregate data generation which is critical to the appropriate decision-making of businesses. The introduction of internet and web technologies enables companies to bring their businesses online. In comparison to traditional businesses, e-businesses have three major advantages. The first one is lower costs for delivering information to customers. For instance, an e-business can inform its customers about recent promotions simply by sending email flyers while a traditional business would need to periodically send printed flyers to its existing customers via conventional postal services. Online businesses can also provide better customer service because they can be contacted via the internet in addition to via phone calls and visits to their physical stores. Moreover, bringing business online gives companies access to the global market and thus greatly increases the potential for business development.

On the other hand, internet and web technologies also pose new threats to businesses. Some of these threats are as follows: formerly disjointed businesses being brought into direct competition; the cost management and efficiencies of a business needing to become competitive on a worldwide basis for the business to survive; and the raising of security and privacy risks as a result of bringing businesses online.

Privacy concerns are a significant factor that e-businesses must take seriously. Many web service users are very cautious about giving out their personal data because most of them have experience in receiving spam email, physical "junk" mail and unsolicited phone calls from marketing companies. Such user fear can ruin e-commerce initiatives.

In this chapter, some of the concepts required to understand the problem definition are briefly explained. Then, the problems in the previously proposed privacy enforcement architecture and its complementary mechanisms are specified. The objectives of this thesis are then illustrated.

1.1 General Concepts:

In this section, some of the concepts required to understand the research context of the thesis are explained. There are five concepts frequently used in the research area of privacy protection and privacy enforcement. They are "Privacy," "The Sticky Policy Paradigm," "Privacy Policy," "Privacy Control" and "Privacy Control Policy." The full definitions of these concepts are as follows:

- Privacy: According to Westin, privacy means an entity's ability to control how, when and to what extent personal information about it is communicated to others [42].

- The Sticky Policy Paradigm: As stated by Karjoth, Schunter and Waidner, the sticky policy paradigm is a concept whereby the users' consent regarding their private data is

associated with their data and "sticks" to the data as it is transferred between different domains [43].

- Privacy Policy: This specifies the privacy promises/practices of an organization regarding the intended uses of the private information collected from users.

- Privacy Control: This pertains to the enforcement of a user's privacy preferences regarding his/her own private data throughout the life-span of the data (collection, retention and transmission, etc.). This can be achieved by using a traditional access control mechanism [1].

- Privacy Control Policy: In this thesis, this is an access control policy that specifies the access control practices of an organization regarding the private information of users.

1.2 Motivation

Privacy is a common concern for e-business users. Privacy violations can be categorized into two major categories. The first is the dissemination of users' private data to other entities without user consent (control lost on extent) [1]. For instance, an e-business could sell the email list of the customers on the market. As the result, the customers will receive large quantities of spam email. The second is abuse of users' private information that has been collected for a legitimate purpose, which implies a loss of control over when and how data is used internally [1]. For example, an e-store can use users' email address and/or residence address which were collected through previous transactions to advertise a new product without user consent.

In Canada, the *Personal Information Protection and Electronic Document Act* (PIPEDA) describes guidelines for the protection of personal electronic information [33]. Principle 7 of PIPEDA describes the safeguards that should be in place to protect sensitive data. The highlights of the aspects of this principle that pertain to my thesis are that:

1. Sensitive information should be protected with a higher level of security;
2. Methods of protection should include technological methods such as the use of passwords and encryption;
3. Limit access to a "need to know" basis;
4. Personal sensitive data should be protected from loss, theft, unauthorized access, disclosure, copying, use or modification.

Apart from PIPEDA, many organizations have put great effort into privacy protection. For example, seal programs have been developed by some companies like TRUSTe to provide a compliant-handling service and to simplify privacy dispute resolution for e-businesses [38]. In another example, the P3P (Platform for Privacy Preferences) policy definition language has been developed by W3C (World Wide Web Consortium) [6]. It allows companies to specify and communicate their privacy policy in a machine readable format. There are two great advantages of P3P policy for web users. The first one is that it forces e-businesses to more precisely specify

their privacy policies in order to alleviate the problem of ambiguity within these policies. The second one is that users can check the P3P policy of a website against their preferences within a second with assistance of client side tools like Privacy Bird from AT&T [23]. Other initiatives include having an external auditor regularly perform privacy impact assessments to ensure that the privacy policies defined/updated by e-businesses are adhered to [1].

A study conducted in 2009 demonstrates that websites' P3P policies do not even follow all the privacy-protection rules mandated by the applicable legislation in force in the governing jurisdiction [46]. It is possible that, even with properly defined P3P privacy policy, an e-business can in practice still use its customers' private data for unintended purposes and there is no effective way to detect and confirm such privacy violations. Despite the efforts in this area by many organizations through technologies like P3P, the automated/effective mechanism for verifying the compliance of e-commerce businesses with their published privacy practices has not yet been developed. As suggested by Adams, it can take a long time and significant resources to resolve a dispute arising from a privacy violation [40].

1.2.1 APEX

In order to comply with PIPEDA and the sticky policy paradigm and thereby ensure organizations' compliance with their published privacy promises, the Architecture for Privacy Enforcement using XML (APEX) was proposed at the theoretical level by Barbieri [1]. The underlying idea is to enforce consistency between XML-based privacy policies specified in P3P and XML-based access control policies specified in XACML by using eXtensible Stylesheet Language Transformation (XSLT) so that the transformation process can be verified by an external auditor. It is therefore possible to ensure consistency between privacy promises and privacy control practices.

There is a major problem with APEX. The whole architecture is based on the assumption that the different kinds of the involved XML-based policies can be *easily* translated from one kind to another kind since they are stated in a common language (XML). The assumption turns out to be false. There are semantic gaps between P3P and XACML. Unless the gaps are properly handled, automated transformation from the one kind of policy to the other kind is impossible. Li and Yan developed an automated transformation mechanism that transforms P3P privacy policies into the corresponding XACML access control policies (section 3.2) [10]. Meanwhile Mahmoudian developed another transformation mechanism that performs the transformation the other way around (section 3.3) [20]. However, none of them performs accurate and effective policy transformation because the semantic gaps between P3P and XACML are not properly handled.

Another problem with APEX is that the use of the User Privacy Preference (UPP) in APEX is not well explained. Neither the UPP language nor the evaluation process is clearly explained [1]. Therefore, APEX cannot effectively address web users' privacy concerns in general.

1.2.2 Policy Based Encryption

Bagga and Molva proposed a Policy Based Encryption (PBE) system that looks very promising for managing access control in a distributed environment [39]. This system provides both encryption and access control in a single package. The PBE system can be used to replace the whole XACML part of APEX in order to achieve privacy enforcement.

1.3 Objectives

In order to address web users' privacy concerns and enforce e-businesses' privacy promises, the main objective of the thesis will be the enhancement of the previously proposed privacy enforcement architecture (APEX). In my thesis research, the enhancement solution of APEX consists of the following elements: (1) the detailed architecture design of EAPEX which handles users' privacy preferences properly so that this design can be followed for implementation; (2) the P3P extension called Privacy Enforcement Extension of P3P 1.1 (P3PPEE) which covers the semantic gaps between P3P and XACML; (3) the accurate mapping of P3P with P3PPEE to XACML so that this mapping can be used by XSLT engine to perform accurate and effective transformation from P3P to XACML in an automated fashion; and (4) the negotiation mechanism that automatically reconciles the contradictions between the privacy promises of e-businesses on the one hand and users' privacy preferences on the other hand.

The secondary objective of my thesis research is to provide the theoretical design of the Architecture for Privacy Enforcement using Policy Based Encryption (APEP).

1.4 Thesis Contributions

This thesis is based on the author's research work in enhancing the previously proposed privacy enforcement architecture and in developing a new privacy enforcement architecture design. In summary, the thesis provides the following major contributions:

1. The detailed architecture design of EAPEX which can be followed for implementation;
2. The extension of P3P which covers the semantic gaps between P3P and XACML and enables the mapping of P3P to XACML;
3. The mapping of the extended P3P to XACML with privacy profile extension that can be used to implement the XSLT transformation from P3P to XACML;
4. The negotiation mechanism for addressing users' privacy concerns;
5. Implementation of EAPEX website and user agent and the illustration of the bandwidth overhead incurred by deploying EAPEX;
6. The theoretical design of the privacy enforcement architecture using the PBE system by Bagga and Molva.

1.5 Thesis Outline

The thesis is organized into six chapters. Chapter 2 provides an overview of existing technologies for privacy protection and the ones for privacy enforcement. Chapter 3 summarizes the related research work on the previously proposed privacy-enforcement framework and on the complementary mechanisms. Chapter 4 describes the user private information that needs to be protected in an e-business environment, the Enhancement of APEX (EAPEX) and the design of the privacy enforcement architecture using the PBE (APEP). It also provides an analysis of the two frameworks. Chapter 5 documents in detail the implementation of the EAPEX e-business website and user agent, as well as the experiments performed on it for measuring bandwidth and time overheads. The thesis concludes with a discussion of the contributions achieved through this research and an overview of research areas and topics related to this thesis that could form the basis for future work.

Chapter 2 Background

Security is vital for online businesses to ensure the authentication and authorization of users and the integrity of content and transactions, as well as to maintain the privacy and confidentiality of sensitive data [21]. In [22], a number of XML-based security standards and applications are discussed. They were proposed with the objective of addressing various specific security issues. For example, the XML vocabulary for sharing security assertions (*e.g.* authentication and authorization assertions) managed by the third party in order to enable "Single sign-on" is defined in the Security Assertion Markup Language (SAML).

Privacy is one important security requirement that limits the access to and use of the user identifiable information that has been collected. A number of organizations have begun efforts to alleviate web users' privacy concerns by developing XML-based privacy standards. An example of the standards is the Enterprise Privacy Authorization Language (EPAL) that is used by enterprises to specify their privacy policies and govern data handling. This chapter introduces the background technologies developed by the well-known organizations W3C (World Wide Web Consortium) and OASIS (the Organization for the Advancement of Structured Information Standards) that are related to my thesis research in the areas of privacy policy specification, privacy preference specification, privacy control policy standards and privacy enforcement mechanisms.

This chapter also introduces several Policy Based Encryption (PBE) systems. The PBE approach can be used as an access control mechanism and thus as an alternative to XACML for achieving privacy enforcement.

2.1 The Platform for Privacy Preferences

Traditionally, organizations publish their privacy promises in a natural language such as English on their websites. Privacy policies specified in natural languages can be misleading, ambiguous, and incomplete, and thus difficult for users to understand. The Platform for Privacy Preferences (P3P) by W3C is an XML-based privacy policy definition language. It allows websites to express their data-collection practices in a standard format that can be automatically retrieved and easily processed by P3P-enabled user agents such as Internet Explorer 8, Netscape 7, and the AT&T Privacy Bird [23] [6]. When a user browses P3P-enabled websites with a P3P-enabled user agent, he will be automatically informed regarding what data will be collected by the websites, what data he may choose to submit or not to submit to the websites, and the dispute-resolution procedures that can be followed in case of a privacy dispute without needing to read the privacy policies at every website that he visits [6] (illustrated in Figure 1). There have been many P3P implementations developed around including user agents/proxies, P3P policy

generators and validators, and server-side P3P support software. Many of them can be found on the web site http://www.w3.org/P3P/implementations.html [12].

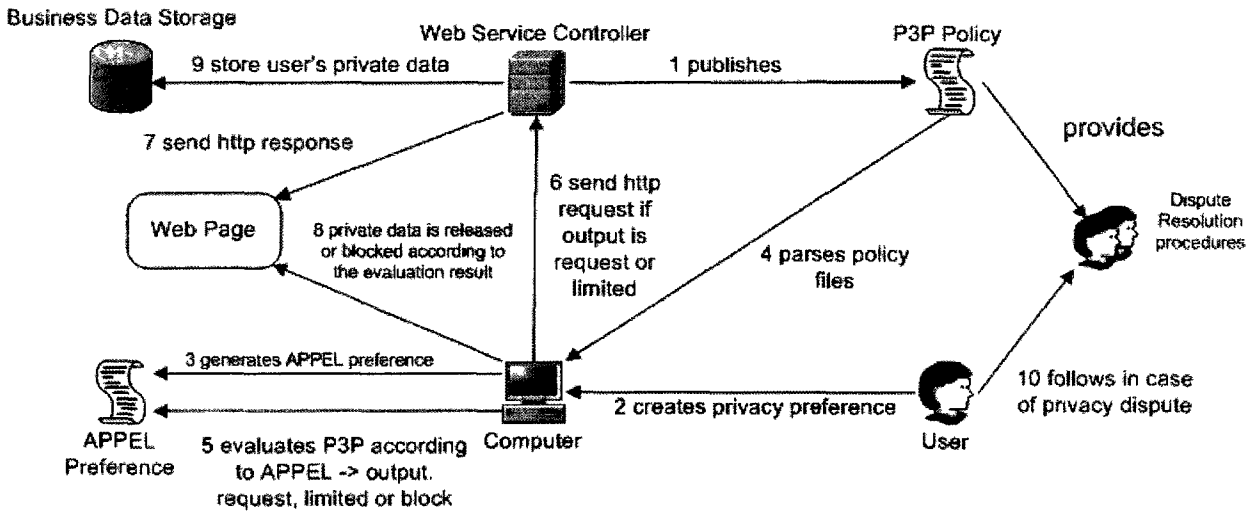


Figure 1 - Http Transactions with P3P Deployed

The current version of P3P is P3P 1.1. It was developed based on the P3P 1.0 specification by adding some new vocabulary pronouns through the P3P 1.0 extension mechanism in order to alleviate the ambiguities of P3P 1.0 [24]. All the new syntax introduced in P3P 1.1 is defined as optional extensions so that P3P 1.1 is fully compliant with P3P 1.0. Thus the semantics of the privacy policies defined in P3P 1.1 will not be misinterpreted by a user agent implemented on the basis of P3P 1.0 specification [24]. The P3P vocabulary includes terms to describe the legal entity making the representation of the privacy practices, the types of private data that the website collects and for what purposes, the organization recipients of the private data collected, the length of time over which the data is retained, the kinds of collected data to which users have access and the dispute-resolution procedures for users to follow if a privacy dispute arises. P3P includes syntax for the policy reference file and a set of privacy policy files. They are explained hereunder [6].

2.1.1 Policy Reference File

In P3P, a policy reference file is an XML file that specifies the URIs where P3P policies can be found, the regions of URI-space covered or not covered by a policy, the cookies that covered by a policy and the time period over which the policy reference file is valid [6]. The following is an example of policy reference file:

```
<META xmlns="http://www.w3.org/2002/01/P3Pv1">
 <POLICY-REFERENCES>
  <EXPIRY max-age="172800"/>
   <POLICY-REF about="/P3P/Policies.xml#browser">
    <INCLUDE>/*</INCLUDE>
    <EXCLUDE>/catalog/*</EXCLUDE>
  ...
  </POLICY-REFERENCES>
</META>
```

It states that P3P policy "/p3p/Policies.xml#browser" applies to the entire site, except to the resources whose paths begin with "/catalog/". The policy reference file (or policy files included) is valid for 172,800 seconds (*i.e.* 48 hours) after retrieval.

2.1.2 P3P Privacy Policy

The P3P privacy policy elements are defined on two levels: the policy level and the statement level. The policy-level elements specify the policy-specific details that are shared by all the privacy statements defined within the policy, including the postal address and contact information of the organization among other details. The statement-level elements describe the actual privacy practices of the organization [6]. All the P3P elements are listed as follows:

The Policy-level Elements:

- <POLICIES>: Used to gather together one or more P3P policies into a single file for the purpose of performance optimization.
- <POLICY>: Used as the container for the other policy-level elements.
- <TEST>: Indicates that the policy is just an example for testing purposes and should not be considered to bea valid P3P policy.
- <ENTITY> (P3P 1.0 and 1.1): Provides a precise description of the legal entity defining the privacy policy. It contains all or some of the fields defined in the business dataset of the base data schema such as the postal address and the telephone number [6]. In P3P 1.1, a <data-group> element which contains <datatype> elements is used through the extension mechanism in addition to the original <DATA-GROUP> defined in P3P 1.0 to describe the hierarchy of the various types of data that provide identification information about the legal entity that runs the website [24].
- <ACCESS>: Describes the capabilities of the individual to access identified data and to address questions or concerns to the service provider.
- <DISPUTES-GROUP>: Groups a set of <DISPUTES> elements.
- <DISPUTES>: Specifies the ways for a user to resolve privacy disputes about the entity's privacy practices. It should contain a <REMEDIES> element.

- <REMEDIES>: Describes the possible remedies that can be applied in case a privacy policy breach occurs.
- <STATEMENT-GROUP-DEF> (P3P 1.1): An optional element used to define an identifier and other optional properties (*e.g.* a short statement group description) that can be applied to a group of <STATEMENT> elements [24].

The Statement-level Elements:

- <STATEMENT>: Used to aggregate any of the disclosures (purposes, recipients and retention) over data elements within this element in order to simplify practice declaration.
- <STATEMENT-GROUP> (P3P 1.1): Describes the statement group with which the statement is associated. Statement groups are used to cluster statements together based on a certain typical usage [24].
- <CONSEQUENCE> (P3P 1.1): Provides a short text summary of the data practices described in the <STATEMENT> element [24]. Note that the definition of this element given here is different from that defined by the P3P 1.0 in which the <CONSEQUENCE> element is used to explain why the suggested practices may be valuable even if the user would not normally allow them [6].
- <NON-IDENTIFIABLE>: Signifies that either only the anonymized data described in the <STATEMENT> element will be collected or no data is collected, if it is included.
- <PURPOSE>: Specifies the purposes for which the data is collected and used.
- <PPURPOSE> (P3P 1.1): A new primary purpose element defined in the P3P 1.1. It provides a more detailed description of data usage [24].
- <RECIPIENT>: Describes the recipients of the collected data.
- <JURISDICTION> (P3P 1.1): Specifies the regulatory environment of the data recipient. This is a new element defined in P3P 1.1. It is included in the <RECIPIENT> element through the extension mechanism [24].
- <RETENTION>: Indicates the type of retention policy that applies to the collected data listed in the <DATA-GROUP> element.
- <DATA-GROUP> (P3P 1.0 and 1.1): contains <datatype> element that describes the type of data that the website collects (P3P 1.1). In order to ensure backward compatibility, the <DATA-GROUP> element of the P3P 1.1 is included through the extension mechanism in addition to this element defined in P3P 1.0 [24].

2.1.3 Backward Compatibility of P3P 1.1

The privacy policies defined in P3P 1.1 are fully compatible with P3P 1.0 since valid P3P 1.1 policies must include all the mandatory elements of P3P 1.0 and the new elements defined in P3P 1.1 through the extension mechanism. On the other hand, in P3P 1.1, some requirements are changed (e.g. < CONSEQUENCE> element) and new definitions are introduced so that the

ambiguity of P3P 1.0 is alleviated without causing any P3P element to have semantic contradiction.

2.2 A P3P Preference Exchange Language 1.0 (APPEL 1.0)

APPEL is a W3C standard that complements P3P 1.0 by providing a language for describing users' preference collections regarding P3P policies. A user can specify his privacy preferences in a set of preference-rules (ruleset) that can be transferred between different P3P enabled user agents. The user can create the APPEL ruleset either manually or by interacting with an APPEL enabled user agent which in turn automatically generates the APPEL ruleset (illustrated in Figure 2). The ruleset can then be used by the user agent or by the user's other APPEL user agents to make automated or semi-automated decisions regarding the acceptability of P3P policies of websites being visited [25] (illustrated in Figure 2).

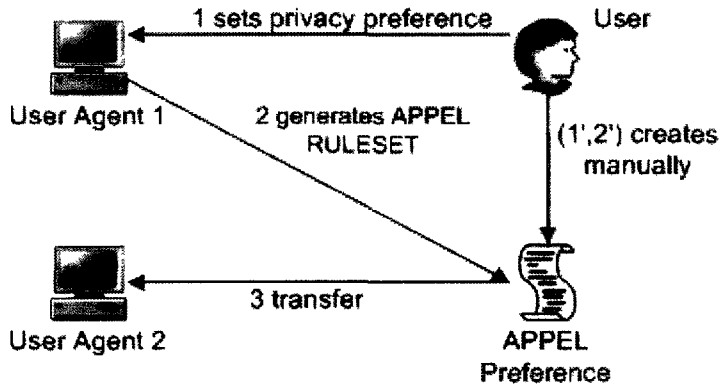


Figure 2 - APPEL

APPEL encoding is consistent with P3P. Most of APPEL's syntax and semantics are influenced by P3P 1.0 since APPEL rulesets are used to specify the user's preferences over P3P polices. APPEL rules are able to prescribe the following set of behaviours: "request," "block" and "limited." When a user visits a P3P enabled website by using a P3P user agent, the user agent fetches the P3P policy that covers the portion of the web site being visited, then uses APPEL rule evaluator to compare the user's privacy preference (rulesets) with the retrieved P3P policy, and finally takes actions based on the behaviour outputs generated by the evaluator [25]. The behaviour outputs are described as follows:

- "request" – the provided P3P policy which covers the URI being visited is acceptable (the resource at the URI should be accessed);

- "limited" – the provided P3P policy is partially acceptable (the resource at the URI should be accessed with limitations);
- "block" – the provided P3P policy is not acceptable (taccess to the resource should be blocked).

Note that the P3P and APPEL do not by themselves provide any mechanism for privacy practice negotiation.

2.2.1 P3P 1.0 Policy Snippet

A P3P 1.0 policy snippet is a simplified P3P policy that may not contain all the required elements specified by P3P 1.0. It is used in an APPEL rule to express users' privacy preference over any set of P3P policy elements. Note that it is not necessary to include all the required elements of P3P policy since the user may not care every part of privacy practices expressed in P3P policy. The absence of a P3P attribute or element in a P3P 1.0 policy snippet allows the attribute/element to be missing from the P3P policy provided by the service or to be included with any value [25]. The wild card symbol "*" can be used to represent the values of certain attributes that the user care. In this case, the user may not care about the values of the attributes but expect them to have some values at least. For example, if "<DISPUTES-GROUP></DISPUTES service="*"></DISPUTES-GROUP>" is included in the P3P snippet of an APPEL rule, it indicates that the user requires the service provider to offer some resolution in case of privacy dispute but he does not care what resolution is provided [25].

2.2.2 APPEL Syntax

In APPEL 1.0, there is an important term called "evidence" which includes a URI and a P3P policy covering the URI. It is used in APPEL rule evaluation. Another critical term is "expression" which is defined as an element of a rule that can be evaluated as being true or false with respect to some evidence [25]. The following is a list of elements and attributes defined in APPEL that are used in addition to those defined in P3P 1.0 in order to create APPEL rulesets.

- <RULESET>: This is the root element of an APPEL file. It contains a sequence of one or more rules.
- <RULE>: Specifies the conditions in the form of expressions and a behaviour output. The behaviour output is returned by the rule evaluator if the expressions included in the rule are all evaluated for their truth relative to the evidence provided. This element has following essential attributes:
    - "behaviour": Specifies the behaviour that should be returned if the rule is evaluated to be true.

    - "prompt": Indicates whether the user is prompted with a message if the rule is evaluated to be true.
    - "promptmsg": States the text message used for prompting the user.
- <OTHERWISE>: This is called the degenerate-expression that can only be included in one <RULE> of the <RULESET> element. The APPEL rule which contains this element is used as a "catch-all" rule and is always placed at the last and evaluated to true,. When the APPEL engine evaluates the <RULESET>, all the rules specified in the <RULESET> are evaluated in the order in which they appear. Once a rule is evaluated to be true, the behaviour output specified in it is returned and rule evaluation ends. If none of the previous rules matches the information contained in the P3P policy and in the URI being covered, then the rule containing the degenerate-expression is triggered and the behaviour output specified is returned.
- <REQUEST-GROUP>: Groups the set of <REQUEST> elements within a single <RULE> element, if multiple alternative domains in form of <REQUEST> elements are specified. The connective "or" or "or-exact" needs to be included to connect them.
- <REQUEST>: Enables the creation of rules that only apply to a particular domain.
- "connective": Describes how the sub-expressions contained at the same inner level are matched while evaluating a rule against the available evidence. If the no "connective" is specified, then "and" is used as the default one.

2.2.3 Rules Ordering

All the rules in an APPEL ruleset are strictly evaluated in order, so there is no need for logical operators between the rules. However, adding a new rule or changing the order of existing rules can significantly influence the rule evaluation result. Therefore, the following rule ordering is suggested by the working group of APPEL. It has been proven to be helpful for both creating and maintaining APPEL ruleset [25].

1. Exceptions (special rules for trusted web sites)
2. Request rules
3. Limited rules
4. Block rules

2.2.4 Simple Ruleset Example

The following APPEL ruleset example states that access to the website that collects information for third parties should be blocked and that access to any other website should proceed with this limitation.

```
<appel:RULESET xmlns:appel="http://www.w3.org/2002/04/APPELv1" xmlns:p3p="http://www.w3.org/2000/12/P3Pv1">
  <appel:RULE behavior="block" description="Service collects data for 3rd parties">
```

```
    <p3p POLICY>
     <p3p STATEMENT>
      <p3p RECIPIENT appel connective="or">
       <p3p same/>
       <p3p other-recipient/>
       <p3p public/>
       <p3p delivery/>
       <p3p unrelated/>
      </p3p RECIPIENT>
     </p3p STATEMENT>
    </p3p POLICY>
   </appel RULE>

   <appel RULE behavior="limited" prompt="yes"
    promptmsg="Suspicious Policy  Do you want to continue (limited access)?">
    <appel OTHERWISE/>
   </appel RULE>
  </appel RULESET>
```

2.2.5 Privacy Bird

Privacy Bird is an Internet Explorer Add-on that supports P3P and APPEL [23]. Firstly, it allows a web user to set his personal privacy preferences, checks if the P3P policies of the website being visited conflict with the user privacy preferences, and notifies the user about the result by means of a bird icon. Secondly, it provides the user with a summary of the website's P3P policies upon user request. The summary contains a bulleted list of the conflict points between the user privacy preferences and the privacy policies; it also includes a summary of the privacy practices specified in the P3P policies. Sherman conducted a survey and found that privacy policies are difficult to understand and that only individuals who have completed at least several years of graduate school education are able to understand the privacy policies of the top Internet companies [44]. Hence privacy bird can certainly help people who are not highly educated. A laboratory study conducted in 2009 found that privacy indicators such as Privacy Bird help people in making decisions regarding online purchases, especially when purchasing privacy-sensitive items [45]. However, an e-business that has properly defined its P3P policies can still in practice abuse its customers' private data. Privacy Bird cannot help web users in this case.

2.3 eXtensible Access Control Markup Language (XACML)

XACML is an OASIS specification which includes a security framework model. It is used for expressing and enforcing distributed, flexible and abstract access control policies using its predefined but extensible taxonomy in XML. Distributed access control policies can be enforced by using it, as different XACML policies specified by different organizations or at different organizational levels can be evaluated together to determine whether or not a particular access request should be granted. XACML allows an organization to specify, merge and analyze rules and policies (sets of rules) according to its unique needs thus achieving flexibility [3] [26] [27].

XACML provides a universal language for authorization policy in order to enable interoperability within a wide range of authorization tools and administrative tools. It allows security policies to be applied consistently across different environments and vendor products.

With the common access control policy mechanism, systems can interoperate with each other thus increasing users' confidence and satisfaction can be more easily developed by companies [3] [26] [27].

2.3.1 The Security Framework Model of XACML

In addition to access control policies, access control decision requests and responses can be expressed in XACML as well. Moreover, a security framework model is included in XACML to achieve access control. This has four important components. They are listed and explained as follows:

- PEP – the policy enforcement point that performs access control. It intercepts a service request, issues the corresponding authorization decision request and enforces the authorization decision received [3] [26] [27].
- PDP – the policy decision point where the authorization decisions are made. Upon receiving an authorization decision request, it evaluates applicable policies from PAP based on the information obtained from PIP such as subject, resource, action and environment attributes. It then renders an authorization decision according to the evaluation result [3] [26] [27].
- PIP – the policy information point which is the source of the information necessary to evaluate the policy. For example, subject resource and/or environment attributes can be obtained from a PIP [3] [26] [27].
- PAP – the policy administration point which is the system entity that is responsible for publishing and managing policies [3] [26] [27].

The ways in which XACML policies are processed in response to a service request in the security framework model are illustrated in Figure 3 below.

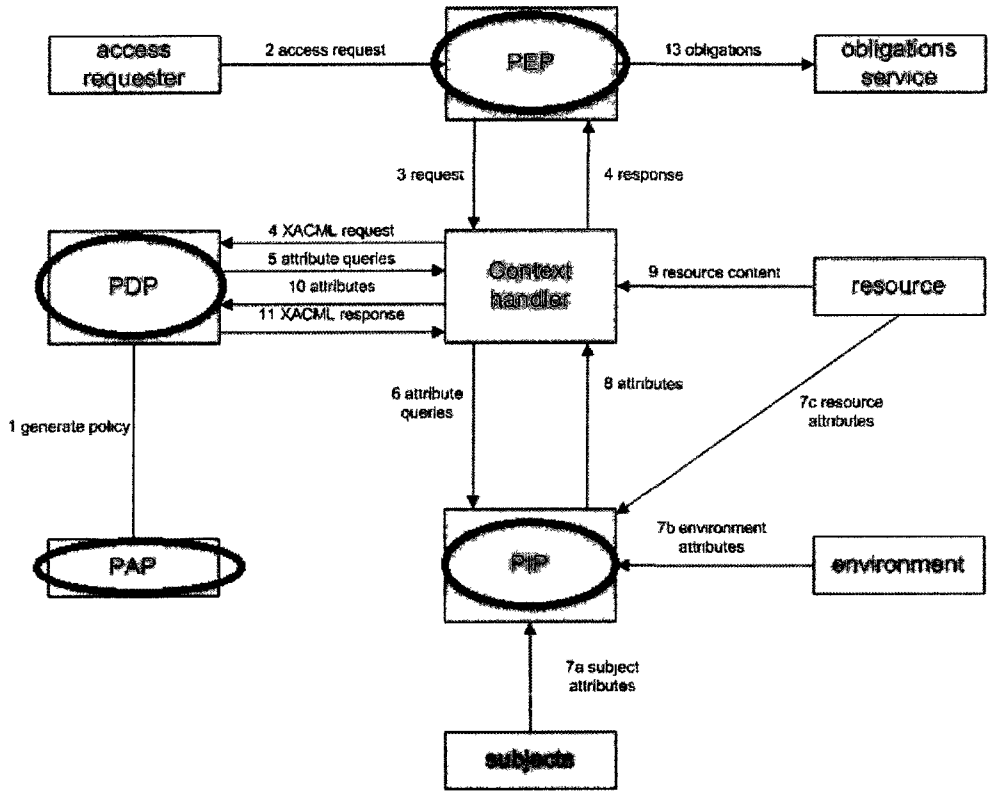


Figure 3 - XACML Data-flow diagram

2.3.2 XACML Policy Language

XACML centers on the following notions:

- <Action>: An operation requested to be performed on a resource [3] [26] [27].
- <Resource>: As defined by the XACML specification, a resource is a data record, service or system component (*e.g.* a printer) [3] [26] [27].
- <Subject>: A user or device which can request to perform an action on a resource [3] [26] [27].
- <Environment>: Any set of attributes that is relevant to making an authorization decision but is not related to a particular subject, resource or action. An example is the current system time of an access request [3] [26] [27].

The three top-level XACML elements are listed and explained as follows:

- <PolicySet>: This element includes a set of policies, other <PolicySet> elements, a policy-combining algorithm and, optionally, a set of obligations. A <PolicySet> is evaluated by the PDP against an access request to render the authorization decision response, if it is applicable to that particular request [3] [26] [27].

- <Policy>: This element contains a set of rules, a rule-combining algorithm and, optionally, a set of obligations. A <Policy> is evaluated against an access request by the PDP, if it is applicable to that particular request. This evaluation result is combined with the evaluation results of other applicable policies by the PDP using the policy-combining algorithm specified in the parent <PolicySet> element to make the authorization decision response [3] [26] [27].
- <Rule>: This element is the primary component of <Policy>. It specifies whether the access requests issued by a set of subjects to perform a certain action on a set of resources are permitted or denied. The evaluation results of applicable rules are combined by using the rule-combining algorithm specified in the parent <Policy> element, and these are in turn combined by using the policy-combining algorithm specified in the parent <PolicySet> element to make the authorization decision response. In addition to the subjects, resources and actions included, a rule may also contain the *condition* under which the rule is applicable. The contained conditions are used to specify certain relationships between the subjects, resources and actions that are targeted by the rule [3] [26] [27].

<Target> and <Obligations> are two other important XACML elements in addition to the ones described above. They are listed and explained as follows:

- <Target>: Defines the decision requests against which the containing rule, policy or policy set is intended to be evaluated in terms of subjects, resources, actions and/or environment attributes involved. The <Target> element contained in a policy or policy set determines the applicability of the policy or policy set to a particular access request respectively. On the other hand, the applicability of a rule is determined by its <Target> and <Condition> elements combined together [3] [26] [27].
- <Obligations>: Specifies the operations of a policy or policy set that should be performed in conjunction with the enforcement of the authorization decision [3] [26] [27].

2.3.3 Privacy Policy Profile of XACML

In the area of XML security standards, the term "profile" means an extension that extends a security standard to achieve a particular goal. XACML 2.0 was developed based on XACML 1.0 by including various profiles and some minor changes in the context and policy schemas. There have been some profiles proposed for XACML 2.0 such as the Hierarchical Resource Profile [28] and the Multiple Resource Profile [29]. Among these, the Privacy Policy Profile of XACML is the one that defines the standard extension for XACML to facilitate its use in privacy applications [30].

The privacy policy profile includes two XACML attributes ("action:purpose" and "resource:purpose") that allow a purpose to be assigned to resources and to actions. When a

purpose is assigned to a resource by using the attribute "resource:purpose", it indicates for what purpose the resource may be used. Similarly, when assigned to an action by using the attribute "action:purpose", the purpose explains why the action is being requested. With this extension, when an action on a particular resource is requested, it will be granted only if the purpose assigned to the action matches the one assigned to the resource [30]. Therefore, privacy can be enforced by using XACML and the privacy policy profile extension.

2.4 The Policy Based Encryption System by Bagga and Molva

Molva and Bagga proposed a Policy Based Encryption (PBE) system based on bilinear pairings over elliptic curves that looks very promising for enforcing access control in a distributed environment where multiple trusted authorities are allowed to participate in the authorization process. As stated in [39], with the PBE system in place, data can be encrypted according to a policy so that only the entities fulfilling the policy can successful decrypt the encrypted data and thus retrieve the plaintext.

According to Molva and Bagga, in general a policy can be formulated either in conjunctive normal form (CNF) or in disjunctive normal form (DNF) because every policy statement in logic consists of a combination of multiple "/\" and "\/". In order to address the two normal forms, policies used in their encryption system are formalized as monotonic logical expressions involving complex disjunctions and conjunctions of conditions in conjunctive-disjunctive normal form (CDNF) [39]. Each condition is defined through a pair (TA, A) in which "A" is an assertion and "TA" is the trusted authority that checks and certifies A's validity.

$$pol = \wedge_{i=1}^{m}[\vee_{j=1}^{m_i}[\wedge_{k=1}^{m_{i,j}}(TA_{i,j,k}, A_{i,j,k})]]$$

In this policy model, m is the number of the clauses of the outer conjunction. Given a specific i (*e.g.* $i = 1$), m_i is the number of the disjunctive clauses of the *ith* outer conjunctive clause. Given a specific pair of i and j (*e.g.* $i = 1$ and $j = 1$), $m_{i,j}$ is the number of the inner conjunctive clauses of the *jth* disjunctive clause of the *ith* outer conjunctive clause. Hence, policies expressed in DNF form are the special cases where $m = 1$, while policies expressed in CNF form are such that $m_{i,j} = 1$ for all i, j.

A policy example is "In order to fulfill this policy the requester must be a member of the International Financial Cryptography Association (IFCA) or the International Chamber of Commerce (ICC), and a full time researcher of company X or a full time analyst of company Y." The monotonic logical expression of the policy in CDNF is: (((IFCA, member) \/ (ICC, member)) /\ (((X, employee) /\ (X, researcher)) \/ ((Y, employee) /\ (Y, analyst)))). The table below illustrates the structure of the logical expression.

i \ j	1	2
1	(IFCA, member)	(ICC, member)
2	((X, employee) ∧ (X, researcher))	((Y, employee) ∧ (Y, analyst))

In this policy example, $m = 2$, $m_1 = 2$, $m_2 = 2$, $m_{1,1} = 1$, $m_{1,2} = 1$, $m_{2,1} = 2$ and $m_{2,2} = 2$.

This policy model allows multiple trusted authorities to take part in the authorization process. As stated by Bagga and Molva, such a policy model is more realistic because each authority should be in charge of a specific, autonomous and limited administrative domain. Moreover, it is more reliable and hence more trustworthy than the policy models relying on a centralized trusted authority to issue the required credentials since the centralized trusted authority could be seen as a single point of failure [39].

A condition (TA_x, A_x) is fulfilled by the credential denoted by $\varsigma(R_x, A_x)$ issued by the trusted authority TA_x using its private key s_x. The credential $\varsigma(R_x, A_x)$ in which R_x is the public key of TA_x is equal to $s_x \cdot Hash(A_x)$. Based on this and on the definition, $\varsigma_{J_1, \ldots, J_m}(pol)$ is defined as the set of $\{\{\varsigma(R_{i,J_i,k}, A_{i,J_i,k}\}_{1 \le k \le m_{i,J_i}}\}_{1 \le i \le m}$, given $j_i \in \{1,\ldots,m_i\}$ for all $i \in \{1,\ldots,m\}$. The policy "pol" is fulfilled if there exists a $j_i \in \{1,\ldots,m_i\}$ for all $i \in \{1,\ldots,m\}$ such that $\varsigma_{J_1, \ldots, J_m}(pol)$ is obtained. The set $\varsigma_{J_1, \ldots, J_m}(pol)$ is used informally to denote the set of credentials that fulfills the policy "pol".

The encryption system proposed in [39] provides a superior and more efficacious way to deal with complex authorization structures than the widely used naive approach which is based on onion-like encryption to handle conjunctions (*i.e.* $pubK_2(pubK_1(m))$) and on multiple encryptions to handle disjunctions (*i.e.* $(pubK_1(m), pubK_2(m))$). The idea of having the encryption system handle the disjunctions contained in the middle of the policy model is explained as follows: each conjunction (*i.e.* inner conjunction of the policy model) of the conditions $\wedge_{i,J} = \wedge_{k=1}^{m_{i,J}}(TA_{i,J,k}, A_{i,J,k})$ is associated with a kind of mask denoted by $\mu_{i,J}$. For each index i, a random chosen key t_i is encrypted (*i.e.* $\oplus$) m_i times using each of the masks $\mu_{i,J}$ to generate the corresponding $v_{i,J}$. As the result, the generated set of $v_{i,J}$ forms a matrix and is sent together with the encrypted message which is encrypted by using $t = \oplus_{i=1}^{a} t_i$. If a user can obtain one instance of the set of credentials $\varsigma_{J_i}(pol)$ and thus compute μ_{i,J_i} for each index i, he/she can retrieve the key t_i for each index i to recover the message. Note that the user needs to use the policy in plaintext in order to determine the indices i and j of the inner conjunction for each of the credentials he/she can obtain with regard to the policy.

A major advantage of this PBE system is that it allows credentials to be kept secret by their owner during policy compliance proofs in contrast to the traditional approach such as XACML in which credentials have to be revealed.

2.5 Other Policy Encryption Systems

Vimercati, Foresti, Jajodia, Paraboschi and Sammarati propose a policy encryption system for achieving access control without involving the data owner [47]. In their system, the encrypted resources and the encryption policy which contains a set of tokens are stored on the server. A resource is encrypted under a particular key. When a user accesses the encrypted resource, he obtains the derivation key from the owner and can retrieve the corresponding token for computing the key only if the user fits into the policy. By using the tokens, the data owner is able to avoid storing all the keys and can fully delegate access control to the server. However, the decryption process of the PBE system relies on the owner for distributing the derivation key. Therefore, it is not feasible to use this system for enforcing privacy in an e-business environment. Bobba, Khurana, AlTurki and Ashraf propose another policy encryption system that allows for access control [48]. But, in this system, the policy used for encryption needs to be kept secret from the recipients. This is not feasible for enforcing privacy in an e-business setting since the employees of an e-business are required to know and understand the privacy policy. Moreover, the PBE system relies on a trusted third party to check the compliance of each user's attributes with the data owner's policy in order to release the decryption key. This is a drawback compared with the PBE developed by Bagga and Molva in which the verification of compliance is automatically achieved.

2.6 Background Summary

This chapter introduced various XML-based privacy technologies developed by W3C and OASIS as well as several PBE systems that can be used as alternatives for access control. However, the missing element is an overall architecture that integrates the technologies to make them cooperate with each other in order to achieve privacy enforcement. The next chapter provides a review of previous research efforts in the areas of privacy enforcement architecture design and its complementary mechanisms.

Chapter 3 Related Work

Web privacy standards specified by using XML work with web content created using XML. The advantages and capabilities of XML (*e.g.* to be extended, combined and adopted widely for a variety of applications and types of content) are preserved in the XML-based privacy standards, so that a common framework (architecture) that addresses various privacy issues can be developed based on the XML-based privacy standards and common XML tools.

The Architecture for Privacy Enforcement using XML (APEX) provides a framework that integrates the XML-based privacy technologies together to enforce organizations' privacy practices. This chapter summarizes and examines the related research work in the areas of privacy enforcement system architecture design and its complementary mechanisms that make the XML-based privacy technologies co-operate with each other.

Policy Based Encryption systems can be used as an alternative to traditional access control systems to enforce privacy. This chapter also summarizes and examines the related research work in privacy enforcement system design based on a PBE system.

3.1 Architecture for Privacy Enforcement using XML (APEX)

The Architecture for Privacy Enforcement using XML (APEX) is a Master's project completed by Barbieri under the instruction of Professor Carlisle Adams at University of Ottawa during academic year 2004 [1]. In this project, Barbieri proposed an overall architecture design that integrates P3P and XACML together to provide a consistent privacy policy enforcement strategy across the enterprise.

As stated by Barbieri in [1], a number of e-businesses, organizations and researchers have begun efforts to alleviate users' privacy concerns. While some e-businesses have begun to publish their textual privacy policies on their websites, others have chosen to use the Platform for Privacy Preferences (P3P) offered by W3C. It provides an XML-based privacy definition language that enables e-businesses to formulate their privacy policies in a standard, machine-readable format. Other initiatives include performing regular privacy impact assessments to ensure that the published privacy policies are adhered to [2]. Privacy promises and audits, however, do not by themselves guarantee to the users that the e-businesses always protect their private data and use it for intended purposes only. In order to achieve privacy policy enforcement, a privacy enforcement architecture design, called the Architecture for Privacy Enforcement using XML (APEX), was proposed in [1] based on existing standards that are gaining acceptance in industry such as P3P and XACML.

3.1.1 Overview of the APEX Architecture Design in [1]

According to Barbieri, privacy control can be achieved by access control which can be implemented by using traditional access control mechanisms. The proposed architecture design is based on XML technologies for the policy specifications (*e.g.* P3P) and on another technology (*i.e.* XACML) to enforce the privacy requirements. The overview of the proposed Privacy Policy Architecture (PPA) is shown in Figure 4.

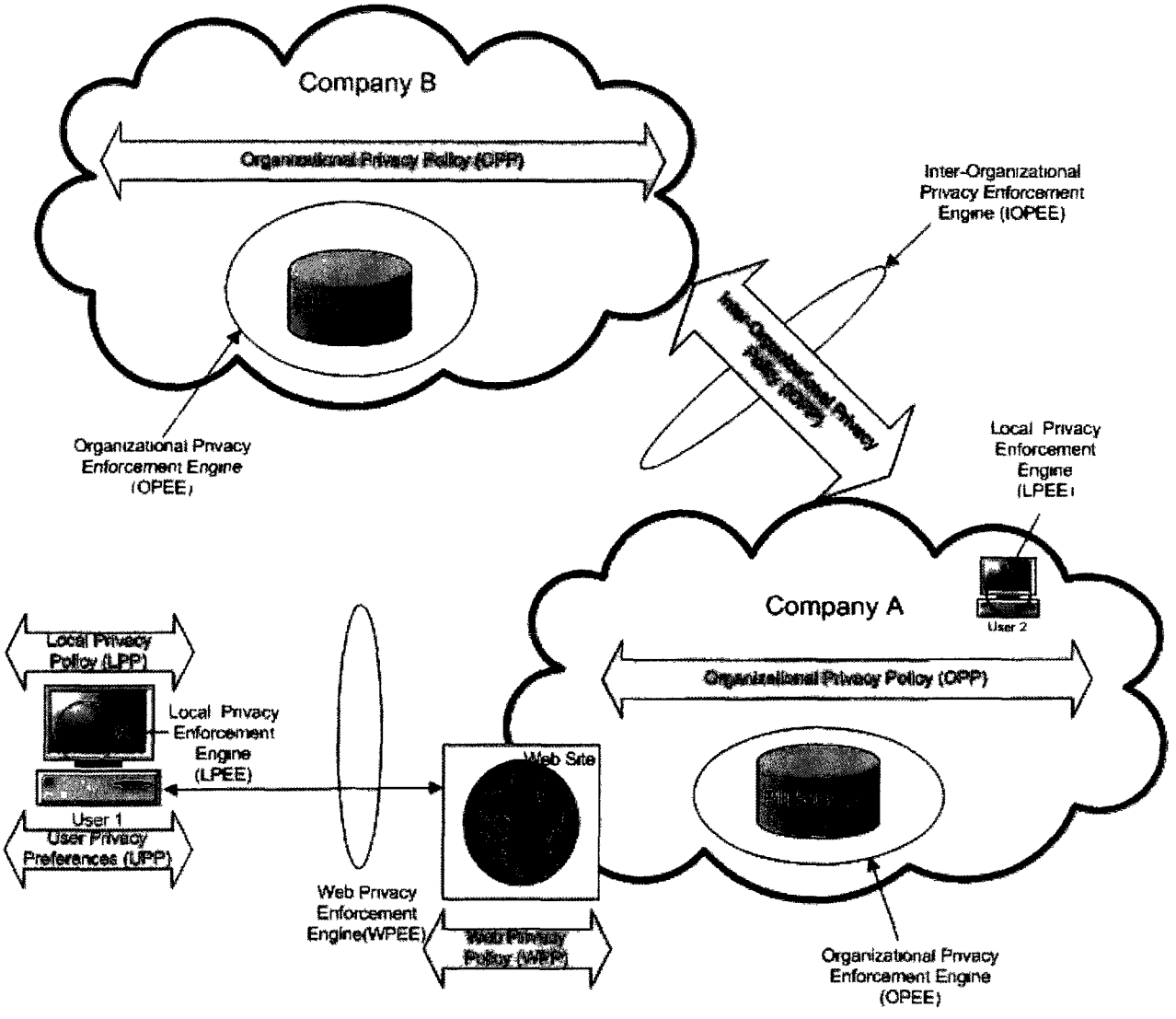


Figure 4 - Privacy Policy Architecture of APEX

As shown in the figure, the different privacy policies required in most organizations at different levels are: Inter-organizational Privacy Policy (IOPP), Organizational Privacy Policy (OPP), Local Privacy Policy (LPP), and Web Privacy Policy (WPP). As stated in [1], these privacy

policies which describe how users' private data is handled within the organization or between organizations are always consistent with the User Privacy Preferences (UPPs). There is, however, no mention of the fact that the mechanism ensures that these policies are always in harmony with the data owners' (web users') privacy preferences.

Several privacy policy enforcement points are also included in PPA. They are: Inter-Organizational Privacy Enforcement Engine (IOPEE), Organizational Privacy Enforcement Engine (OPEE), Local Privacy Enforcement Engine (LPEE), and Web Privacy Enforcement Engine (WPEE). These privacy enforcement points are used to ensure that each access to users' private data at different levels complies with the appropriate privacy policies. The PPA is derived from transformation of the company-wide access control policy using Extensible Stylesheet Language Transformation (XSLT) engines [1].

The Privacy Enforcement of PPA is achieved by the Privacy Enforcement Architecture (PEA) (illustrated in Figure 5) which is partially based on the XACML architecture [1][3].

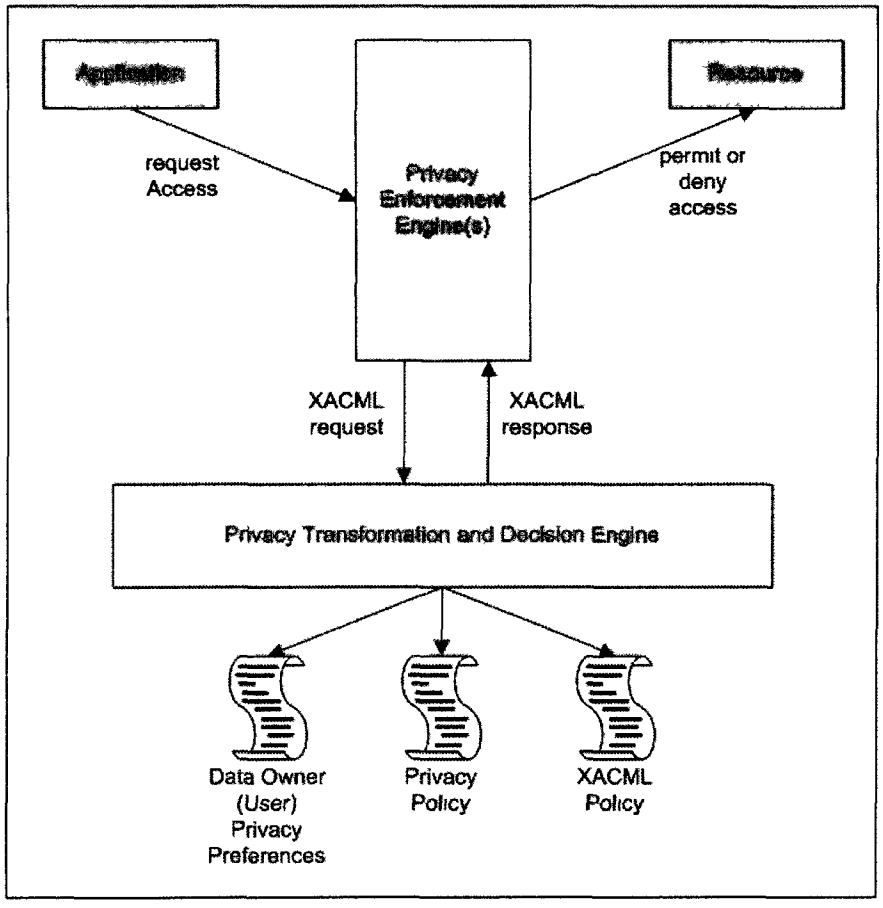


Figure 5 - Privacy Enforcement Architecture

In PEA, the request handling is similar to the one in XACML. The Privacy Enforcement Engine (PEE) is deployed at key choke points in the enterprise to process access requests. When an access request is sent to the PEE, it is formulated in the XACML syntax and is forwarded to the Privacy Transformation and Decision Engine (PTDE). The PTDE is responsible for transforming any applicable XML-based privacy policies to XACML access control policies and then evaluates the request against the applicable XACML access control policies to produce an XACML response which includes the action to be enforced and possibly some obligations. Then the PTDE sends the XACML response to PEE. Finally, PEE permits or denies the access requested according to the XACML response received [1].

The Privacy Policy Architecture and the Privacy Enforcement Architecture which are described above together form the APEX architecture design explained in [1]. As stated by Barbieri, the assurance that the privacy policies are properly enforced is greatly enhanced by using privacy enforcement points and audited transformation of policies to maintain consistent privacy control decisions across the enterprise [1].

3.1.2 Privacy Architecture Components

3.1.2.1 XML-based Privacy Policy

In APEX, only four XML-based privacy policies deployed at different levels are used throughout the organization. They are LPP, WPP, OPP and IOPP. Additional XML-based policies that are applicable in specific environments can be included [1].

3.1.2.1.1 Local Privacy Policy (LPP)

The rules to be enforced while transacting private data on the local system are specified in the Local Privacy Policy [1]. Researchers have developed two approaches for governments and other trusted organizations to derive personal (or local) privacy policies templates in a semi-automated manner [4].

3.1.2.1.2 Web Privacy Policy (WPP)

The rules to be enforced while transacting private data on the web are specified in the Web Privacy Policy [1]. P3P is a popular XML-based privacy policy language that websites can use to specify WPP [6][5].

3.1.2.1.3 Organizational Privacy Policy (OPP)

The rules to be enforced while transacting private data within the organization are specified in the Organization Privacy Policy (OPP) [1]. EPAL [7] is used for defining OPP in APEX.

3.1.2.1.4 Inter-Organizational Privacy Policy (IOPP)

The rules to be enforced while transacting private data between organizations are specified in the Inter-Organizational Privacy Policy [1]. Although EPAL is chosen to define IOPP in APEX, it is

not complete as the researchers identified some requirements that are not covered by the current version of EPAL (v1.2) [8].

3.1.2.2 Privacy Transformation and Decision Engine (PTDE)

The Privacy Transformation and Decision Engine is composed of the Transformation Engine (TE) and the Decision Engine (DE) [1]. Figure 6 shows how the XML-based privacy policies are transformed to or derived from XACML access control policies by the Privacy Transformation Engine (PTE).

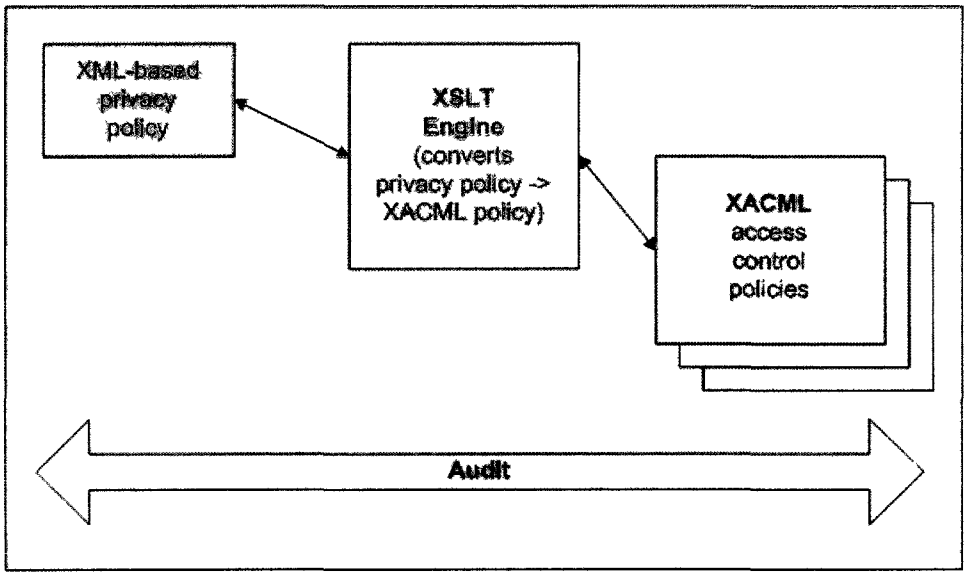


Figure 6 - Privacy Transformation Engine

There are several possible functions for the TEs. They transform the organization's XACML access control policy to the OPP and WPP whose privacy promises are reflective of their actual access control practices. They are also responsible for transforming other XML-based policies (*e.g.* user privacy preferences) into the corresponding XACML policies that can be used by the decision engine to compare with web and organizational privacy policies in order to make an XACML response [1]. There is a problem with the transformation from UPP into the corresponding XACML. The technology used in APEX to specify UPP is APPEL, although it is not clearly stated in [1]. The UPP specified in APPEL which contains only a P3P policy snippet can be very abstract in the sense that it does not have enough semantics to be transformed into the corresponding XACML policy.

The major function of the decision engine is to apply policy-combining algorithms to make an XACML response upon receiving an XACML request. The combining algorithms can be used to specify the rules regarding how to combine applicable policies. The list below shows the order in which the decision engine evaluates applicable policies. Since all the privacy policies in APEX are transformed to XACML, the PDP of XACML can be used as the decision engine [1]. [11] provides an implementation example of the PDP of XACML.

1. user (data owner) preferences
2. enterprise privacy policies (LPP, WPP, OPP, IOPP)
3. traditional access control policies (XACML)

3.1.2.2.1 Transformation Engines

The research conducted by IBM [9] provides an approach to transform EPAL policies into P3P policies using mapping tables which describe the mapping of the different elements of the two policy languages. In APEX, a similar method of using mapping tables is employed in the transformation engines to transform policies into the corresponding policies in a different policy language. The TEs of APEX refine the method proposed in [9] by automating the transformation using XSLT engines. For example, the APEX transformation engines use XSLT to translate XML-based privacy policies (*e.g.* P3P) to – or to derive them from – XML-based access control policies (*e.g.* XACML). Different transformation engines are required for transforming different XML-based privacy policies (*e.g.* P3P, EPAL) [1].

The XSLT transformation engine which converts XACML policies to P3P policies have previously been researched and implemented by two students at the University of Ottawa [10]. The transformation mechanism is illustrated in Figure 7. This will be explained in detail in section 3.2.

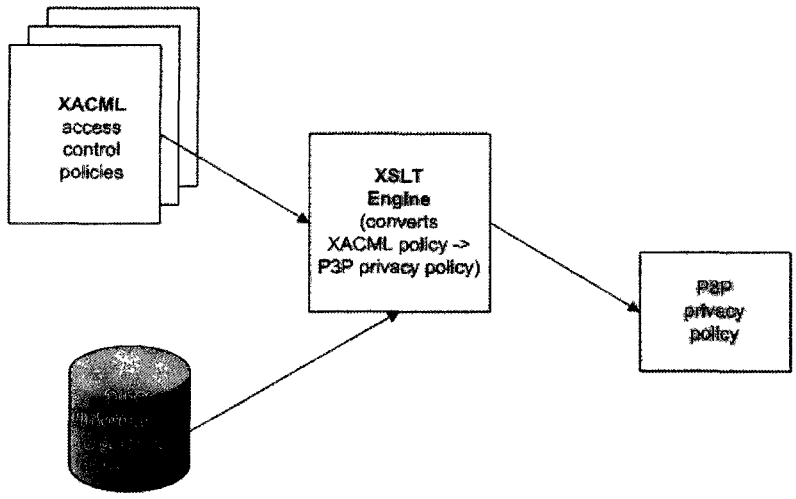


Figure 7 - XSLT Engine for XACML to P3P

3.1.2.3 Privacy Enforcement Engine(s) (PEE)

Access control decisions with obligations will be enforced by PPEs in their respective domains [1].

3.1.2.3.1 Web Privacy Enforcement Engine (WPEE)

The Web Privacy Enforcement Engine which can be implemented in the web user agent (*e.g.* internet browser) or in a web proxy is used to enforce privacy policies at a web point of presence for activities of an e-commerce type such as the submission of private data [1]. An APPEL engine [12] is a form of WPEE that extracts the P3P privacy policies of the website being visited and then checks them against the user's privacy preferences. It allows the user's private data to be sent out only when the P3P privacy policy matches the preference. Conversely, it prompts the user or blocks the http request.

3.1.2.3.2 Organizational Privacy Enforcement Engine (OPEE)

Access responses that correspond to the access requests for private data throughout the organization are enforced by OPEE. OPEE can be implemented in the file server's file system, as a mandatory application proxy or gateway [1].

3.1.2.3.3 Inter-Organizational Privacy Enforcement Engine (IOPEE)

Access responses that correspond to the access requests for private data between organizations are enforced by IOPEE. IOPEE can be implemented in a manner similar to that of OPEE [1].

3.1.2.3.4 Local Privacy Enforcement Engine (LPEE)

Access control for all local access to private data is enforced by LPEE [1]. Two possible implementations of LPEE were researched at the HP Trusted Systems Laboratory. One is to attach privacy labels or tags to the private data as it traverses the system. The other one makes use of Identifier Based Encryption (IBE) to encrypt private data in local transactions thus enforcing privacy control [13].

3.1.2.4 PII Storage

In APEX, the PII storage component is a privacy-aware database or other data storage device that stores all the private data collected. The PII must be able to co-operate with privacy enforcement engines to ensure that only authorized access to private data is allowed. Thus the traditional database security features such as encryption, passwords and auditing should be included in PII [1]. The Hippocratic Database which was discussed in [14] is one example of a technology to provide privacy enforcement at the database level. A Hippocratic Database takes privacy into consideration by including "purpose" as a basis for data retrieval; thus, compared with a traditional database, it focuses more on consented sharing but less on efficiency. Implementation of a Hippocratic Database can be based on the technologies used for preserving

privacy in statistical databases. In [15], the same researchers suggested that a Hippocratic Database can be easily integrated with P3P by mapping P3P <PURPOSE> and <RETENTION> to "purpose" in order to enforce the privacy promises stated in a P3P policy. Another example of a privacy-aware database is the pawDB introduced in [17].

3.1.2.5 Audit

Two types of auditing are included in APEX. The first type is comprised of the audit logs containing all the access requests regarding users' private data and their corresponding response actions. This type of auditing is performed by Privacy Enforcement Engines, and the audit logs should be kept in a well-protected environment. The second type is comprised of external auditing that is used to provide additional assurance for the organization's privacy enforcement. For example, a trusted organization can perform a privacy assessment that focuses on the PEE, the TEs and the DE of an APEX-enabled organization [1]. The technologies that can possibly be used to implement auditing include database-level audit logs, self-certifying software [18], and external certification initiatives (*e.g.* the Common Criteria [15]), which can be used to provide privacy certifications and Privacy Seal Programs (e.g. TRUSTe [16]) [1].

3.1.2.6 Summary Table of APEX Components and Current Technology or Research

APEX Component	Current Technology or Research
Local Privacy Policy	Research at NRC [4]
Web Privacy Policy	P3P policies [6]
Organizational Privacy Policy	EPAL polices [7]
Inter-Organizational Privacy Policy	EPAL policies [7] Efficient Comparison of Enterprise Privacy Policies [16]
Privacy Transformation Engine	XSLT Engine for transformation from XACML to P3P [10] Transforming EPAL to P3P [9] XSLT Engine for other transformations
Privacy Decision Engine	XACML PDP engine [11]
Web Privacy Enforcement Engine	APPEL implementation [12] Implementing P3P Using Database Technology [15]
Organizational Privacy Enforcement Engine	File server's file system Mandatory application proxy Internal gateway that intercepts all access to stored private data
Inter-Organizational Privacy Enforcement Engine	File server's file system Mandatory application proxy Extranet gateway that intercepts all access to stored private data
Local Privacy Enforcement Engine	HP Trusted System Laboratory [13]

PII Storage	Hippocratic Databases [14] pawDB [17]
Audit	Operating System Logs Network Logs Application Logs Database Logs Self-Certifying Software [18] Common Criteria Certifications [15] Privacy Seal Program

Table 1 – Summary of APEX Components and Technologies

3.1.3 Issues and Problems of APEX

3.1.3.1 Issues of APEX

Several issues of APEX have been discussed by Barbieri in [1]. Some of them that are related to my thesis research are explained and examined in the following sub-sections.

3.1.3.1.1 Privacy Laws and Regulations: Jurisdictions

According to Barbieri, legislation, policy and technology together comprise privacy. The legislation is not handled by the design of APEX. Organizations must define their privacy polices based on the applicable privacy legislation. The privacy policies' compliance with legislation should be somehow ensured [1]. P3P version 1.1 has begun to incorporate "jurisdiction" by including a <JURISDICTION> element in the extension of <RECIPIENT> element. It allows an organization to specify the jurisdiction with which a particular data recipient is complying [1]. A possible solution is that a trusted party would perform a privacy assessment on the organization to check its privacy policies (*e.g.* P3P policies) against the privacy legislation specified in P3P <JURISDICTION > whenever the policies are modified.

3.1.3.1.2 Dealing with Private Data after Retrieval

After the successful retrieval of private data, the data is in an unprotected state. The relevant privacy policy can be violated if the data is transmitted to an unauthorized user or system by a malicious user. The LPEE included in APEX can be used to detect such a violation and prevent it. But the technical implementation could be challenging [1].

3.1.3.1.3 Purpose

The purposes of the collection of private data cannot be specified in traditional access control systems [19]. The solution proposed in [1] is to extend XACML by adding "purpose" as a specific action attribute in the XACML schema for APEX. However, checking the purpose of an action requested on private data can be bypassed if the request issuer is able to include the appropriate value of "action-purpose" in his/her request. For example, the marketing employees of an organization can set the value of "action-purpose" to be <P3P: delivery/> in order to access

users' postal information for marketing purposes. This problem is addressed by the <PURPOSE-ROLES> element of the extension for P3P 1.1 which is introduced in section 4.2.2 of this thesis.

3.1.3.1.4 Standard Privacy Policy Vocabularies

The successful implementation of a privacy enforcement engine depends on standardized or at least compatible vocabularies for different policy specifications (*e.g.* P3P and XACML). Although some research has been done in this area, the implementation of mapping language vocabularies is still challenging because the semantic differences between the access control policy language (XACML) and the privacy policy specification (P3P) cannot be easily handled [1]. The extension for P3P 1.1 introduced in section 4.2.2 is proposed to handle the semantic gaps between P3P and XACML. This extension extends P3P specification through the extension mechanism defined in P3P 1.1 so that the RBAC models of organizations can be easily included in P3P.

3.1.3.2 Problems of APEX Not Discussed in [1]

In the section, the problems of APEX that are not discussed by Barbieri but are found in my thesis research are discussed and examined.

3.1.3.2.1 Using LPP as UPP

The research conducted at the National Research Council (NRC) [4] provides two approaches to derive personal (or local) privacy preferences in a semi-automated fashion. The trusted organizations or the communities of peers can apply these approaches to create user privacy preference templates for different uses of the data (*i.e.* different online services). Such User Privacy Preferences (UPPs) are not Local Privacy Policies (LPPs) because the former define the user's preferences with regard to transactions on private data between different domains (*i.e.* between users and organizations). Therefore, LPPs should not be included in the set of policies that is evaluated by the PTDE since it specifies only the rules to be followed while transacting with private data on the local system.

3.1.3.2.2 The Multiple Transformation Engines

All the transformations from XACML policies to the corresponding policies in different XML-based privacy languages or vice versa can be achieved by a single XSLT engine with different XSLT mappings. In addition, for the sake of the simplicity and efficiency of APEX, there should be only one XSLT engine which is the Privacy Transformation Engine.

3.1.3.2.3 The Insufficient Semantics of UPP Specified in APPEL

As mentioned earlier in the summary of the APEX architecture design, the User Privacy Preferences (UPPs) specified in APPEL can be very abstract since they contain only P3P policy snippets. Thus they do not have enough semantics to be transformed into the corresponding effective XACML policies which are used by the PTDE. So web users' privacy concerns cannot

be addressed using the APEX architecture design proposed in [1]. The negotiation mechanism for EAPEX proposed in section 4.2.4 can be used to solve this problem. The mechanism abstracts the contradicting parts of the WPP according to UPP. Then it transforms the parts into valid XACML policies/rules on the client side, and submits the generated XACML policies/rules together with the user private data to the website.

3.1.3.2.4 Irrelevant Parts of the UPP

As stated by Barbieri in [1], the UPPs submitted with the users' private data are transformed into corresponding XACML access control policies (privacy control policies) by the privacy transformation engine in APEX. As explained in section 3.1.3.2.3, such a transformation is not even possible. If it were feasible, this process would cause unnecessary performance problems since it would transform the relevant parts as well as irrelevant parts of the UPP. Moreover, each resulting privacy control policy generated from irrelevant parts of the UPPs would downgrade the performance of the decision engine unnecessarily, thus making the performance problems even worse. Thus, only the relevant parts of the privacy preferences that contradict the organization's WPP should be submitted to the organization and be passed to the PTDE. This problem is resolved by the negotiation mechanism for EAPEX explained in section 4.2.4

3.1.3.2.5 The Absence of the Common Privacy Preference Template Set

In [4], "the common privacy preference template set" is defined. It can be used by both the web users and the organizations with APEX deployed to alleviate the conflict between the users' privacy preferences and the organization's privacy policies. The number of the relevant parts of the UPPs transmitted to PTDE can thereby be reduced which, in turn, can alleviate the performance downgrade problem.

3.1.3.2.6 The Time of the Transformation of the UPPs

Whether the users' privacy preferences should be transformed by PTE immediately after the transmissions or should be processed by PTE in batches is not discussed in [1]. Immediate transformations cause performance problems on the organization side while batch processing may cause unintended privacy violations. This problem is solved by the negotiation mechanism proposed for EAPEX in section 4.2.4. This negotiation mechanism functions in an alternative way since the transformation of the parts of the WPP that contradict the UPPs are performed on the client side.

3.2 Automated Translations for Architecture for Privacy Enforcement using XML (APEX)

Automated Translations for APEX [10] is an honours project completed by Dongyi Li and Hui Yan under the instruction of Professor Carlisle Adams at the University of Ottawa during academic year 2004-2005. In this project, Dongyi Li and Hui Yan developed an automated

translation mechanism which transforms a set of XACML access control policies to the corresponding P3P privacy policies. This section summarizes the mapping pattern of their mechanism and discusses several major problems within their research work in [10]. The full analysis of the mapping which describes all the problems found is included in Appendix 3. In this section the automated translation mechanism developed by them is called the Automated Transformation from XACML to P3P (ATXP).

According to Li and Yan, XACML is relevant to privacy control because it is the mechanism that corporations use to guard customer information from unintended use, and thus it reflects their actual privacy practices. An e-business's XACML access control policies contain information such as which party has access (read or write) to what kind of data collected from the customers and which party does not have access to what kind of data. Therefore the P3P policies of the e-businesses' websites can be derived from their XACML access control policies so that the privacy practices stated in the resulting P3P policies are consistent with the corporation's access control practices. Moreover, given the fact that both P3P and XACML are XML-based languages, it follows that the transformation from XACML policies to the corresponding P3P polices can be performed automatically by using Extensible Stylesheet Language Transformation (XSLT) engines [10].

3.2.1 Data Format

Since P3P and XACML focus on different aspects of privacy control, information that needs to be specified in some P3P policy elements may not be included in the XACML access control policy. Additional information that covers the gap between P3P and XACML must be included in the data resource and in pre-defined files in order to ensure the success of the transformation from XACML to P3P. The data file used in this project is an XML file which includes multiple <Records> elements. Each <Records> element is identified by its "subject-id" attribute which refers to the owner's id. Each <Records> element includes multiple <Record> elements in each of which a particular group of user information can be specified. Each <Record> element has two attributes – "id" which identifies the record and "task-purpose" which indicates the use purpose for which the user's private information is collected and stored in the record. A <Record> element contains one <Labels> element in which various kinds of missing information for the P3P policy can be specified by the corresponding types of labels. A <Record> element also contains a <Data-group> element which describes the kinds of user private information contained in the record. The format of <Data-group> is the same as the one in P3P [10].

3.2.1.1 Data Resource Example

The following is an example of a <Records> element that demonstrates the missing information contained in a data resource for a P3P policy.

```
<Records subject-id="Paul" xmlns="urn:example:schemas:records">
 <Record id="ID" task-purpose="admin">
  <Labels>
   <purpose category="develop"/>
   <purpose category="individual-analysis" opt-in="yes"/>
   <purpose category="other-purpose" opt-out="yes">checking</purpose>
   <recipient category="ours"/>
   <recipient category="same" opt-in="yes"/>
   <recipient category="delivery" opt-out="yes"/>
   <Subject-access category="contact"/>
   <DATA-GROUP>
    <DATA ref="#user.name " />
    <DATA ref="#user.home-info.postal " />
   </DATA-GROUP>
  </Labels>
  <Customer>
   <name>Paul Simpson</name>
   <address>...</address>
   ...
  </Customer>
 </Record>
</Records>
```

3.2.2 The Three Key Mapping Patterns and Details [10] [3] [6]

The following subsection explains the mapping pattern for extracting information from XACML policies to generate the corresponding P3P <ACCESS> element in detail. After that the mapping patterns for the other P3P elements covered by ATXP are briefly explained in section 3.2.2.2.

3.2.2.1 Mapping Pattern for the P3P <ACCESS> Element [10]

As defined in P3P 1.0, the <ACCESS> element describes the capability of private data owners (*i.e.* customers) to view the identified data collected from them and to address questions or concerns to service providers (*i.e.* e-businesses). This element must contain one of the following elements:

<nonident/> The website does not collect identified data.

<all/> Access is given to all identified data.

<contact-and-other/> Access is given to identified online and physical contact information as well as to certain other identified data.

<ident-contact/> Access is given to identified online and physical contact information.

<other-ident/> Access is given to certain other identified data (*e.g.* users can access things such as their online account charges).

<none/> No access to identified data is given.

Since the customers of an e-business are the owners of the private data collected from them, they should be able to perform both read and write (*e.g.* update their contact information) actions on the private data collected from them. Thus all the relevant rules specified for the owner with "action=read/write" and "effect=permit/deny" in a policy are checked for the following patterns:

1) Checking Step

Condition 1: Only the policies written for "owner" will be processed further. The related code fragment in the condition element of a rule element of such a policy to check is as follows:

```
<Apply FunctionId="string-equal">
 <AttributeValue>owner</AttributeValue>
 <Apply FunctionId="string-one-and-only">
  <SubjectAttributeDesignator AttributeId="subject:subject-category"/>
 </Apply>
</Apply>
```

Condition 2: For every XACML <Policy> element which contains rules under condition 1, the values of the category attributes of the subject-access element referred to by the rules which meet condition 1 are checked. For each such rule contained in the <Policy> element, if the subject-access is found to be equal to "nonident" it is marked as case 1; if the subject-access is found to be equal to "other" it is marked as case 2; and if the subject-access is found to be equal to "contact" it is marked as case 3.

Case 1:

```
<Apply FunctionId="string-equal">
 <AttributeValue>nonident</AtributeValue>
 <Apply FunctionId="string-one-and-only">
  <AttributeSelector RequestContextPath="/n1:Record[@id='ID'/Labels/Subject-access@category='nonident']"/>
 </Apply>
</Apply>
```

Case 2:

```
<Apply FunctionId="string-equal">
 <AttributeValue>other</AtributeValue>
 <Apply FunctionId="string-one-and-only">
  <AttributeSelector RequestContextPath="/n1:Record[@id='ID'/Labels/Subject-access@category='nonident']"/>
 </Apply>
</Apply>
```

Case 3:

```
<Apply FunctionId="string-equal">
 <AttributeValue>contact</AtributeValue>
 <Apply FunctionId="string-one-and-only">
```

```
  <AttributeSelector RequestContextPath="/n1:Record[@id='ID'/Labels/Subject-access@category='nonident']"/>
 </Apply>
</Apply>
```

2) Aggregating Step

In an XACML policy written for the private data owners, all the rules whose condition elements have a code fragment matching one of the three cases and meeting condition 1 are aggregated. If all the rules checking case 2 have the deny effect and at least one permit rule checks case 3, then the value for the resulting P3P <ACCESS> element is <ident-contact/>. If all the rules checking case 3 have the deny effect and at least one permit rule checks case 2, then the result is <other-ident/>. If at least one permit rule checks case 3 and at least one checks case 2, then the result is <contact-and-other/>. If all the rules checking either case 3 or case 2 are deny rules, then the result is <none/>. If all the rules checking either case 3 or case 2 are permit rules then the result is <all/>. If at least one permit rule checks case 1 and none of the rest checks either case 2 or case 3, the resulting <ACCESS> element value is <nonident/>.

3.2.2.2 Mapping Patterns [10] [3] [6]

Similarly, information extracted from XACML policies which contain rules checking whether the requested resource is a customer private data record is aggregated to generate the other elements of P3P policies. This section addresses the details of XACML policy modeling and the extracting patterns.

1) <P3P:POLICIES>: Each XACML <PolicySet> element is used to generate the corresponding <P3P:POLICIES> element.

```
<PolicySet PolicyCombiningAlgId=""permit-overrides>
 <Target>
 ......
 </Target>
 ..policy 1
 ..policy 2
 ......
</PolicySet>
```

2) <P3P:EXPIRY>: The bolded XACML code checks the request time against a designated time. This designated time is used to generate the P3P <EXPIRY> element of the corresponding P3P <POLICIES> element.

```
<PolicySet>
 <Target>
  <Subjects><anySubject/></Subject>
  <Resources><anyResource><Resources>
```

```
  <ActionMatch MatchId="date-less-and-equal">
   <AttributeValue>05-01-2004</AttributeValue>
   <Action AttributeDesignater AttributeId="Environment:current-time"/>
  </ActionMatch>
 </Target>
 ..policy 1
 ..policy 2
 ......
</PolicySet>
```

3) <P3P:POLICY>: Every XACML <Policy> element whose <Target> elements check whether the requested resource is a customer private data record and has the designated record id is transformed into the corresponding P3P <POLICY> element.

4) <P3P:STATEMENT>: The rules written for "nonowner" which grant only read access to customer records for users other than the private data owner are transformed into the corresponding P3P <STATEMENT> elements.

5) <P3P:CONSEQUENCE>: The information contained in <Description> elements of rules is extracted to generate the corresponding P3P <CONSEQUENCE> elements.

6) <P3P:PURPOSE>: The information for generating the contained element of the P3P <PURPOSE> of the P3P <Statement> which corresponds to the selected XACML <Rule> element is extracted from the following code pattern contained in the <Rule> element:

Case 1: <P3P:develop/>

```
<Apply FunctionId="string-equal">
 <AttributeValue>develop</AttributeValue>
 <Apply FunctionId="string-one-and-only">
  <ActionAttributeDesignator AttributeId="action:action-purpose"/>
 </Apply>
</Apply>
```

Case 2: <P3P:individual-analysis/>

```
<Apply FunctionId="and">
 <Apply FunctionId="string-equal">
  <AttributeValue>individual-analysis</AttributeValue>
  <Apply FunctionId="string-one-and-only">
   <ActionAttributeDesignator AttributeId="action:action-purpose"/>
  </Apply>
 </Apply>
 <Apply FunctionId="string-equal">
  <AttributeValue>yes</AttributeValue>
  <Apply FunctionId="string-one-and-only">
   <AttributeSelector
   RequestContextPath="/Records/Record[@id='ID']/Labels/purpose[@category='individual-analysis']/@opt-in"/>
  </Apply>
 </Apply>
```

```
</Apply>
```

Case 3: <P3P:other-purpose>checking</P3P:other-purpose>

```
<Apply FunctionId="and">
 <Apply FunctionId="string-equal">
  <AttributeValue>other-purpose</AttributeValue>
  <Apply FunctionId="string-one-and-only">
   <ActionAttributeDesignator AttributeId="action:action-purpose"/>
  </Apply>
 </Apply>
 <Apply FunctionId="string-equal">
  <AttributeValue>yes</AttributeValue>
  <Apply FunctionId="string-one-and-only">
   <AttributeSelector
   RequestContextPath="/Records/Record[@id='ID']/Labels/purpose[@category=other-purpose']/@opt-in"/>
  </Apply>
 </Apply>
 <Apply FunctionId="string-equal">
  <Apply FunctionId="string-one-and-only">
   <AttributeSelector
   RequestContextPath="/Records/Record[@id='ID']/Labels/purpose[@category='other-purpose']/text()"/>
  </Apply>
  <Apply FunctionId="string-one-and-only">
   <ActionAttributeDesignator AttributeId="action:action-specific"/>
  </Apply>
 </Apply>
</Apply>
```

Case 4: <P3P:current/>

```
<Apply FunctionId="string-equal">
 <Apply FunctionId="string-one-and-only">
  <AttributeSelector RequestContextPath="/Records/Record[@id='ID']/@task-purpose/>
 </Apply>
 <Apply FunctionId="string-one-and-only">
  <ActionAttributeDesignator AttributeId="action:action-purpose"/>
 </Apply>
</Apply>
```

The rule pattern that aggregates the results of action-purpose checking is as follows:

```
<Apply FunctionId="or">
 case1
 case2
 case3
 case4
</Apply>
```

The Resulting P3P <PURPOSE> element is as follows:

```
<PURPOSE>
 <current/>
 <individual-analysis required="opt-in"/>
 <other-purpose required="opt-out">checking</other-purpose>
```

</PURPOSE>

7) <P3P:RECIPIENT>: The XACML rule code pattern of the mapping for P3P <RECIPIENT> element is similar to the code pattern for the mapping of P3P <PURPOSE> element. The information is extracted from the XACML code contained in the applicable rules' condition that checks if the attribute "subject-category:subject-recipient" has the value of "ours", "same" or "delivery".

8) <P3P:RETENTION>: The information for generating P3P <RETENTION> elements is extracted from the code pattern contained in applicable rules which checks the issue date of the access request against the expiry date specified in the requested data record element.

9) <P3P:DATA-GROUP>: The mapping pattern for P3P <DATA-GROUP> locates an applicable rule/policy which checks whether the requested resource is a customer private data record. The resource XPath expression embedded within the applicable rule/policy then locates the record file and extracts the <DATA-GROUP> element from the record file.

10) <P3P:NON-IDENTIFIABLE>: The mapping pattern checks whether the code fragment for generating a P3P <RECIPIENT> element or <PURPOSE> element is missing in an applicable rule. If this is the case, the mapping pattern generates a P3P <NON-IDENTIFIABLE> element for the P3P <STATEMENT> element that corresponds to the applicable rule.

3.2.2.3 P3P Elements Not Covered by XACML [10] [3] [6]

The information for generating P3P elements <ENTITY> and <DISPUTE> cannot be specified in XACML; thus such information should be extracted from the pre-defined files of the company.

3.2.3 The Major Problems with the Mapping of ATXP

This section briefly describes the two main problematical mapping patterns of ATXP found in my thesis research. The analysis that explains all the problems found is included in appendix 3.

3.2.3.1 The Mapping Pattern for <P3P:PURPOSE>

The attribute identifiers "action:action-purpose" and "action:action-specific" used in the mapping of ATXP are specified neither in the project documentation nor in the XACML specification 1.0 . This problem is caused by the semantic differences between P3P 1.0 and XACML 1.0. In the

privacy policy profile of XACML 2.0, similar privacy related attribute identifiers are defined and can be used to address the problem.

3.2.3.2 The Mapping Pattern for <P3P:RECIPIENT>

The code that checks the organization of which the requester is an employee (hereinafter "requester's organization") should be included in a <SubjectMatch> element of the <Target> element of an applicable rule. In the example shown in section 4.1.1 of the XACML 1.0 specification, the matching code pattern should check whether the requester's "subject-id" attribute (*i.e.* the requester's email address) matches the company domain, the delivery company domain, or a company that follows equivalent practices. Thus the mapping should extract information from the XACML code described above to generate the P3P <RECIPIENT> element. If the information of the recipients is available in the data record of a certain type (*e.g.* ID type) then it can be directly extracted from the data record instead of from the applicable XACML policies. There is another minor mapping problem with the mapping patterns. The problem is that the attribute identifier "subject-category:subject-recipient" used in the mapping pattern is neither specified in the XAML specification nor explained in the document. This problem is caused by the semantic difference between P3P 1.0 and XACML 1.0.

3.2.4 Contribution of ATXP in Terms of Privacy Control

ATXP provides an efficient means to automatically transform a company's access control policies specified in XACML to the corresponding P3P privacy policies. However, it imposes many constraints on customers' private data records such as the requirement that the <DATA-GROUP> element must be included, and the constraints on the format to specify the XACML policy customized for generating P3P policies that are used to partially cover the semantic differences between P3P 1.0 and XACML 1.0. Therefore, the usefulness of ATXP is restricted.

3.3 Developing an Internal Access Control Policy for a Website Using an Automated Privacy Policy Mapping

Developing an Internal Access Control Policy for a Website Using an Automated Privacy Policy Mapping [20] is an honours project completed by Mahmoudian under the instruction of Professor Carlisle Adams at the University of Ottawa during academic year 2004-2005. In this project, Mahmoudian developed an automated translation mechanism using XSLT which converts P3P privacy policies to the corresponding XACML access control policies. This section summarizes the mapping pattern used in the mechanism and discusses several of its main

problems which are caused by the semantic differences between P3P and XACML. The full analysis of the mapping which discusses all the problems found within the research work is included in Appendix 4. In this section the automated translation mechanism developed by Mahmoudian is called the Automated Transformation from P3P to XACML (ATPX)

The Platform for Privacy Preferences (P3P) and the eXtensible Access Control Markup Language (XACML) were the two XML-based security standards mainly discussed in [20]. There are many organizations that have for many years used P3P to specify their website's privacy policies in order to address growing privacy concerns and regulations. However, privacy concerns cannot be addressed by using only P3P since the privacy management tool that would convert a privacy policy statement into an access control policy and would thereby enforce the privacy practise is not provided in P3P. An implementation of this tool which uses an XSLT to transform P3P to XACML was developed by Mahmoudian in [20].

3.3.1 Relationship between P3P and XACML

According to Mahmoudian, the P3P and XACML are compatible with each other and serve complementary purposes. The main objective of P3P is to express privacy policies in a form that can be processed by computers, while the objective of XACML is to express privacy control policies in a form that enables them to be enforced by computer systems. In P3P, privacy policies are expressed at a high level of generality of user and data categories. In XACML, privacy control policies are expressed in terms of specific data resource identities or of resource descriptors assigned by the system. The expressions in the two languages are theoretically compatible. Hence, according to Mahmoudian the XACML standard is a concrete application of the P3P standard to actual users, resources, actions and purposes [20]. An example of using P3P and XACML together is a privacy rule made by a clinic to comply with provincial patient privacy rules. It states that "the protected health information of a patient must be disclosed only to the patient or the patient's representative." Even though the requirement is expressed in generic terms in a P3P policy, the corresponding XACML policy can make it concrete by listing the specific information fields in specific computer files that qualify as "protected health information." The match between the accessing user's authenticated identity and the relevant fields in the requested health record file is required to permit access to the health information record [20]. As it is shown in the example, while externally published policies expressed in P3P are in a generalized, high-level form, policies expressed in XACML are in a fine-grained internally applicable form. Therefore, the two levels of policy together would enable an auditor to determine whether the enterprise's stated privacy policies are being complied with [21].

On the other hand, there is a gap between P3P and XACML in that the mechanism for ensuring that the websites adhere to their stated privacy policies is not provided in P3P and in that a

method of directly and automatically converting from P3P to XACML did not exist at the time that Mahmoudian completed his research. As stated in [20], the mechanism that automatically maps P3P policies to the corresponding XACML policies is beneficial to the organizations performing auditing and to those implementing privacy and access control policies.

3.3.2 The Motivation of Automated Privacy Policy Mapping

XACML can provide privacy control that enforces the privacy promises throughout the organization. However, the mechanism that effortlessly and automatically converts a privacy policy (*i.e.* P3P) into an XACML policy had not yet been developed at the time of Mahmoudian's research [20]. The absence of such a mechanism formed the motivation for his research work.

3.3.3 Mapping Implementation

According to Mahmoudian, mandatory elements of XACML standards must be distinguished and obtained from relevant elements of P3P specification. Thus, deploying an automated privacy policy mapping mechanism should consist of the following steps [20]:

1. Mapping the data types defined in the P3P privacy policies to the corresponding data resource identifiers or system-assigned resource descriptors used in XACML;
2. Mapping the data users defined in the P3P privacy policies to enterprise roles defined in the access control mechanism of the enterprise;
3. Based on the privacy promises made, deriving an access rule for the private data of each group of users in terms of subjects and conditions;
4. Complementing the resulting access control policy with any further details needed in order to obtain a well-defined XACML policy.

According to Mahmoudian, the privacy practices regarding the related private data should be included in a single <STATEMENT> element when the organization is creating P3P privacy policies so that the resulting XACML equivalent is a single <Rule> as opposed to a large number of <Rule> elements. If the P3P policies are created as described above, then the efficiency of processing the resulting XACML polices derived from the P3P by using the Privacy Policy Mapping increases as the number of resulting <Rule> elements decreases.

3.3.3.1 Mapping Patterns

1) <P3P:POLICY>
A complete P3P privacy policy file contains a single <POLICIES> element which itself contains one or more <POLICY> elements [6]. In [20], each <POLICY> is mapped to an equivalent XACML <Policy> element. The value of the attribute "PolicyId" of the resulting <Policy> element is derived from the attribute "name" of the <POLICY> element [20].

2) <P3P:EXPIRY>
The <EXPIRY> element of a P3P policy indicates an expiry date and time until which the given privacy policy is valid. Although there is no equivalent element in XACML for direct mapping, the meaning of <EXPIRY> can be captured by an equivalent function in the <Condition> element. This function makes a permit rule applicable only if the issue date and the time of access requests are less than or equal to the designated ones. The following XACML code fragment shows the resulting function for a given <EXPIRY> element [20].

P3P Input: <P3P:EXPIRY date="Tue, 16 Dec 2002 12:10:10 GMT">

XACML output:

```
<xacml:Condition FunctionId="and">
  <xacml:Apply FunctionId="date-less-or-equal">
    <xacml:Apply FunctionId="date-one-and-only">
          <xacml:EnvironmentAttributeDesignator
          DataType=http://www.w3.org/2001/XMLSchema#date
          AttributeId="environment:current-date"/>
    </xacml:Apply>
  <xacml:AttributeValue
          DataType="http://www.w3.org/2001/XMLSchema#date">2002-12-16</xacml:AttributeValue>
  </xacml:Apply>
  <xacml:Apply FunctionId="time-less-or-equal">
    <xacml:Apply FunctionId="time-one-and-only">
          <xacml:EnvironmentAttributeDesignator DataType="http://www.w3.org/2001/XMLSchema#time"
          AttributeId="environment:current-time"/>
    </xacml:Apply>
    <xacml:AttributeValue
          DataType="http://www.w3.org/2001/XMLSchema#time">12:10:10-5:00</xacml:AttributeValue>
  </xacml:Apply>
</xacml:Condition>
```

3) <P3P:ACCESS>
The P3P ACCESS element indicates the kinds of the collected identifiable data to which the users have access. The information contained in the P3P <ACCESS> element can be used to deduce an XACML policy which is applicable to identifiable data owner [20]. The function checks the requester's identity as is shown in the following:

```
<xacml:Condition FunctionId="and">
  <xacml:Apply FunctionId="string-equal">
    <xacml:AttributeValue
        DataType="http://www.w3.org/2001/XMLSchema#string">owner</xacml:AttributeValue>
    <xacml:SubjectAttributeDesignator AttributeId="string-one-and-only"
        DataType="http://www.w3.org/2001/XMLSchema#string"    SubjectCategory="access-subject"/>
  </xacml:Apply>
</xacml:Condition>
```

In [20], the following types of resource categories are derived with respect to the users' identified data:

Case 1:

```
<xacml:Apply FunctionId="string-equal">
  <xacml:AttributeValue   DataType="http://www.w3.org/2001/XMLSchema#string">nonident</xacml:AttributeValue>
  <xacml:ResourceAttributeDesignator
      AttributeId="resource-location" DataType="http://www.w3.org/2001/XMLSchema#string"/>
</xacml:Apply>
```

Case 2:

```
<xacml:Apply FunctionId="string-equal">
  <xacml:AttributeValue   DataType="http://www.w3.org/2001/XMLSchema#string">other</xacml:AttributeValue>
  <xacml:ResourceAttributeDesignator
      AttributeId="resource-location" DataType="http://www.w3.org/2001/XMLSchema#string"/>
</xacml:Apply>
```

Case 3:

```
<xacml:Apply FunctionId="string-equal">
  <xacml:AttributeValue   DataType="http://www.w3.org/2001/XMLSchema#string">contact</xacml:AttributeValue>
  <xacml:ResourceAttributeDesignator
      AttributeId="resource-location" DataType="http://www.w3.org/2001/XMLSchema#string"/>
</xacml:Apply>
```

Depending on the type of <ACCESS> element value specified in the P3P policy, the types of resources in the resulting XACML policy to which the user has access are defined as follows [20]:

<nonident/>
    Permit case 1 and deny both case 2 and case 3.
<ident-contact/>
    Permit case 3 and deny case 2.
<other-ident/>
    Permit case 2 and deny case 3.
<contact-and-other/>
    Permit both case 2 and case 3.
<all/>

Permit case 1, case 2 and case 3.

<none/>

Deny both case 2 and case 3.

If the P3P policy contains the following ACCESS element:

```
<ACCESS>
 <ident-contact/>
</ACCESS>
```

Then the equivalent XACML code fragment would be as follows:

```
<xacml:Condition FunctionId="and">
  <xacml:Apply FunctionId="string-equal">
    <xacml:AttributeValue
          DataType="http://www.w3.org/2001/XMLSchema#string">owner</xacml:AttributeValue>
    <xacml:SubjectAttributeDesignator AttributeId="string-one-and-only"
          DataType="http://www.w3.org/2001/XMLSchema#string"    SubjectCategory="access-subject"/>
  </xacml:Apply>

  <xacml:Apply FunctionId="string-equal">
    <xacml:AttributeValue   DataType="http://www.w3.org/2001/XMLSchema#string">contact</xacml:AttributeValue>
    <xacml:ResourceAttributeDesignator AttributeId="resource-location"
          DataType="http://www.w3.org/2001/XMLSchema#string"/>
  </xacml:Apply>

  <xacml:Apply FunctionId="not">
  <xacml:Apply FunctionId="string-equal">
    <xacml:AttributeValue   DataType="http://www.w3.org/2001/XMLSchema#string">other</xacml:AttributeValue>
    <xacml:ResourceAttributeDesignator
          AttributeId="resource-location" DataType="http://www.w3.org/2001/XMLSchema#string"/>
  </xacml:Apply>
  </xacml:Apply>

</xacml:Condition>
```

4) <P3P:STATEMENT>

The P3P <STATEMENT> allows the organization to group collected user data by its category and to explicitly state which parties have access to it and for what purposes. The <STATEMENT> element is a container that includes a <PURPOSE> element, a <RECIPIENT> element, a <RETENTION> element, a <DATA-GROUP> element and, optionally, a <CONSEQUENCE> element [6]. The <STATEMENT> container allows organizations to group together data elements that are handled the same way and to create a distinct <STATEMENT> element for each group [6]. Each distinct <STATEMENT> element describes a specific type of user identifiable data to be collected and identifies unique subject groups that are allowed to access the collected data. Thus, the specifics of each distinct <STATEMENT> element can be translated into a distinct XACML <Rule> element. The output of mapping all <STATEMENT> elements contained in the same <POLICY> element is a single XACML <Policy> element that contains the same number of <Rule> elements as are required to allow appropriate access

decision-making based on the types of data being requested and on the domain of the requester's ID. Since all <STATEMENT> elements refer to the read access of users' private data, the resulting <Policy> element should have the following overall Target groups [20]:

<Subjects> → <AnySubject/>
<Resources> → user_data
<Actions> → read

Each <STATEMENT> element is in turn read to extract relevant information for generating the <Resource> and <Subject> elements for the resulting applicable <Rule> elements within the <Policy>.

5) <P3P:DATA-GROUP>
The P3P <DATA-GROUP> element specifies the categories (or types) of user identifiable data to collect, so that the data categories of <DATA> contained in it are used as resource-id in the <Target> element of the resulting <Rule> [20].

6) <P3P:RECIPIENT>
The P3P <RECIPIENT> element indicates the recipients of the private data collected in terms of the website being visited and any involved third party. It contains one or more of six possible recipient categories: <ours>, <same>, <other-recipient>, <delivery>, <public> and <unrelated>. The most restrictive is <ours> while <unrelated> is the least restrictive category that allows access by any other organization to the private data collected from users [6]. Since one or more recipient categories can be included in a P3P <RECIPIENT> element, the string bag function is used in the resulting XACML code fragment. Upon receiving an access request, the subject category is checked against the categories contained in the string bag function. If there is a match, the access is granted [20].

P3P input:

```
<RECIPIENT>
 <ours/>
 <other-recipient/>
</RECIPIENT>
```

XACML output:

```
<xacml:Condition FunctionId="string-at-least-one-member-of">
        <xacml:Apply FunctionId="string-bag">
                <xacml:AttributeValue
                        DataType="http://www.w3.org/2001/XMLSchema#string">ours
                </xacml:AttributeValue>
                <xacml:AttributeValue
                        DataType="http://www.w3.org/2001/XMLSchema#string">other-recipient
                </xacml:AttributeValue>
        </xacml:Apply>
```

```
        <xacml:SubjectAttributeDesignator AttributeId="string-one-and-only"
                DataType=http://www.w3.org/2001/XMLSchema#string
                 SubjectCategory="subject-category:access-subject"/>
</xacml:Condition>
```

10) Rule Combining Algorithm

The Deny Overrides Rule Combining Algorithm was used in [20] to reconcile the decisions made by evaluating the rules in the <Policy> as it was stated by Mahmoudian that, in the resulting XACML policy, one rule should not be allowed to circumvent other rules [20].

3.3.3.2 Unusable P3P Elements

In [20], the following P3P elements were determined not to be applicable in the mapping because the information contained in them is not transferable to an access control policy: <ENTITY>, <DISPUTES>, <CONSEQUENCE>, <PURPOSE>, and <RETENTION> [20].

3.3.4 The Major Problems with the Mapping of the ATPX

This section briefly describes the two main problems of the mapping patterns of ATPX found in the research for this thesis. The analysis that explains all the problems of ATPX discovered over the course of this research is included in appendix 4.

3.3.4.1 The Mapping Pattern of <P3P:RECIPIENT>

In the mapping pattern of the <P3P:RECIPIENT> element, the attribute "subject-category" and the attribute value "access-subject" cannot be used to determine the organization of the requester in terms of the recipient group involved. Moreover, the "string-one-and-only" is a function id and thus cannot be used as an attribute id. The following XACML code fragment should instead be used to check the domain of the requester's id (*e.g.* if the requester's email address, "someone@companya.com" is indeed equal to "CompanyA").

```
<Target>
 <Subjects>
  <Subject>
   <SubjectMatch
   MatchId="urn:oasis:names:tc:xacml:1.0:function:rfc822Name-match">
    <AttributeValue
    DataType="http://www.w3.org/2001/XMLSchema#string">
    CompanyA.com
    </AttributeValue>
    <SubjectAttributeDesignator
    AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
    DataType="urn:oasis:names:tc:xacml:1.0:data-type:rfc822Name"/>
   </SubjectMatch>
```

```
  </Subject>
 </Subjects>
</Target>
```

However, the XACML code fragment set out above cannot itself completely address the problem because there is no explicit element in P3P that expresses the connections between the organizations of the requesters and the organization recipient categories. Thus, the resulting XACML policy does not contain the code – as shown in the above code fragment – to check the requester's organization when only the recipient categories are given. This problem is caused by the semantic difference between P3P 1.0 and XACML 1.0. A possible solution is to use <recipient-description> to specify the respective URLs of the organization recipients included within the recipient category tags. However, the use of <recipient-description> is optional according to P3P specifications 1.0 and 1.1 [6] [24]. This problem is solved by the Privacy Control Extension of P3P proposed in section 4.2.2.

3.3.4.2 The Absence of the <PURPOSE> Element in the Mapping

The intended uses of the private data collected are expressed in the <PURPOSE> element. Such information should be captured and somehow transformed into the corresponding organization's RBAC model that can be integrated into the resulting XACML access control policies to achieve privacy enforcement. The <PURPOSE> element is, however, not included in the mapping of ATPX. For example, an organization states a privacy promise in its P3P policy that users' postal address information is collected and used to complete users' online orders only. But even with the solution described in 3.3.4.1, the XACML polices derived by using ATPX will check only the organizations of the request subjects and thus may allow marketing employees to access users' postal address information for marketing purposes which is a contravention of the privacy promise. This is an obvious privacy violation that is not prevented by the resulting XACML policies derived by using the ATPX. This problem is caused by the semantic difference between P3P and XACML. It is addressed by the mapping introduced in section 4.2.3 which is based on the privacy enforcement extension introduced in section 4.2.2 of this thesis.

3.3.5 The Contribution of ATPX in Terms of Privacy Control

According to Mahmoudian, the Automated Privacy Policy Mapping developed in [20] allows any P3P privacy policy to be automatically translated into its equivalent machine enforceable XACML access control policy. It is claimed that this automated mechanism enables privacy enforcement because the privacy practises stated in the P3P policies can be efficiently converted to the corresponding access control practices and enforced by XACML. It is claimed that ATPX can also be used to perform auditing because auditors can use this mechanism to generate the

corresponding XACML polices and to compare them with the actual XACML polices deployed in organizations in order to ensure privacy enforcement.

On the other hand, it is clear that the usefulness of ATPX is highly limited due to the problems described above and to those included in appendix 4. The general mapping logic and its mapping patterns are correct, but the mechanism cannot be used in practice unless all the problems are properly addressed.

3.4 Security and Privacy System Architecture for an e-Hospital Environment

In [41], Garson and Adams proposed a theoretical overall system architecture called the Mobile Emergency Triage (MET) for ensuring the security and privacy of patients' medical records in an e-hospital environment. The major component of the MET system for privacy enforcement is the extended Privacy Based Encryption schema which was originally proposed in [39].

3.4.1 The e-Hospital Environment

According to Garson and Adams, many hospitals that currently use paper-based systems for patients' medical records are migrating towards electronic medical records. Electronic record systems provide automated processes and computer applications that can help doctors with the diagnosis and treatment of patients. Hospitals using electronic record systems can provide better care for their patients. Nevertheless, deploying the systems raises security and privacy risks with regard to patients' sensitive information. Once the patient records in electronic format are compromised, a malicious individual can easily collect large amounts of sensitive data. Hence electronic patient record systems must have security and privacy measures in place to ensure proper authorization and confidentiality for records and thereby achieve compliance with privacy statutes PIPEDA and PHIPA.

3.4.2 The Mobile Emergency Triage (MET) System

In [41], Garson and Adams propose a theoretical system architecture called the Mobile Emergency Triage (MET) system for managing access to patients' records and for ensuring the security of the records in transmission on the network. The MET system is designed to support doctors when making evidence-based decisions in an emergency room setting by retrieving relevant information from a collection of medical literature and electronic health records. The major component of the MET system for privacy enforcement is the extended version of the

Policy Based Encryption (PBE) system proposed in [39]. The MET system also consists of the electronic health record system and tablet PCs with the MET software to help doctors in their tasks. It works on a wireless network and thus enables doctors to move from patient to patient while having access to the software.

3.4.2.1 The Technology Alternatives

The technology used in the MET system for privacy enforcement is the Policy Based Encryption (PBE) system developed by Bagga and Molva in [39]. The technology alternatives for adding security and privacy functionality to the MET system are examined and discussed in [41]. They are network encryption & access control, public key cryptography, identity/role-based encryption and policy-based encryption. The eligibilities of using these technology alternatives in the MET system are explained as follows:

Network Encryption & Access Control: Role-based access control provides protection against unauthorized access while network encryption provides security for sensitive data during transmission. The MET system can be implemented based on the combination of the two mechanisms. However, they were not determined to be the best option due to the complexity of combining them [41].

Public Key Cryptography: In a public key system, a plaintext is encrypted for a single recipient using his/her public key. Thus, it cannot be used to encrypt medical records for access by multiple recipients (*e.g.* the doctors in the hospital). In addition, key management is another factor that limits its use in the MET system [41].

Identity/Role Based Encryption: The Identity Based Encryption (IBE) builds on the idea of encrypting under an arbitrary string. By using an IBE system, the need for key management is eliminated because a user's public key is a well-known unique string (e.g. *his/her* email address) and the user obtains the private key by authenticating to the trusted authority. However, IBE cannot be used to encrypt for multiple recipients. Role Based Encryption (RBE) is a more general approach than IBE since the former allows records to be encrypted for a general grouping such as a role. But it cannot be used to encrypt for multiple roles. For example, it is a common practice in a hospital setting that doctors and nurses have access to patient records, and therefore patient records need to be encrypted for both doctors and nurses. It is obvious that RBE systems are not suitable in this case [41].

Policy Based Encryption (PBE): According to Garson and Adams, the idea behind PBE is the generalized idea of IBE. If the public key can be an arbitrary string, then this string can be a complete access control policy or a hash of it instead of an identity. Each document type has an associated access policy and a corresponding decryption key. A user authenticates himself by

logging into the system and thus receives the decryption keys associated with his role. Then, when the user accesses a record, if the user's role satisfies the policy his corresponding decryption key will decrypt the document. The database in which the encrypted records are stored needs to have access to decryption keys and policies in order to index the records.

Since with the least amount of complexity the PBE approach ensures both security and privacy of sensitive data and thus provides the most compliance with the PIPEDA legislation, PBE is considered to be the most promising technology alternative and is used in the MET system architecture [41].

3.4.2.2 The Extension to the Policy Based Encryption System Developed by Bagga and Molva.

The PBE system developed by Bagga and Molva is chosen to be used in the MET system architecture. Garson and Adams proposed an extension to this PBE system to add more flexibility for use in the e-hospital environment [41].

The PBE system works well for a policy that specifies one action for a specific resource. But the policies need to be applied to two actions (*i.e.* read and write) in the e-hospital environment. Garson and Adams extend the policy model of the PBE system by allowing actions to be specified in conjunction with other rules in a policy. An example of a policy is "doctors can read and write a certain type of documents, and nurses can only read the type of documents." The policy can be expressed as using the policy model with the extension pol = (((role:doctor) /\ (action:write)) \/ (role:nurse)). By using "read" as the default value for actions, both doctors and nurses can perform the read action on the type of documents. But only employees who satisfy the other rule in the conjunction can perform the write action.

3.4.3 The Major Problems with the MET System Architecture

In this section, the problems of MET system that are not discussed in [41] but are found in my thesis research are discussed and examined.

3.4.3.1 The Difference between the PBE System Explained and the One Used in MET

The PBE system described in [41] is different from the one developed by Bagga and Molva. The characteristics of the former are listed as follows: a policy itself or a hash of it is used for encrypting a certain type of document; the corresponding decryption key is kept secret by the Private Key Generator (PKG); if a user successfully authenticates himself to the system and his role satisfies the policy, he will obtain the decryption key for that type of document; and the

functionality of the system for determining whether the user's role (*i.e.* an attribute) satisfies the policy is exactly the same as the one of a traditional access control system like XACML.

One the other hand, in the PBE system developed by Bagga and Movla, a randomly generated key is used for both encrypting and decrypting a specific document. The key consists of randomly generated pieces. Each piece is encrypted multiple times using each of the masks of the associated disjunctive clauses of the policy. When a user requests the document, the data owner sends to the user the encrypted document and the policy as well as the matrix formed by the masks in which the key is hidden. If the user's credential set (*i.e.* the attribute assertion signed by the trusted third parties specified in the policy) satisfies the policy, then he can retrieve the key by computing all the pieces of the key and then using the key to decrypt the encrypted document [39]. It should be noted that, unlike the PBE system described in [41], the verification of the user's compliance with the policy is automatically achieved in the PBE proposed in [39].

3.4.3.2 Patient Record Indexing

As stated in [41], the PBE system developed by Bagga and Movla is used in the MET system. According to Garson and Adams, the database used in the MET system must have access to decryption keys as well as to the policies for indexing records. However, in order to comply with the PBE schema, the encrypted document, the policy under which the document was encrypted and the matrix in which the key is hidden all need to be stored together in the database. The key for decrypting a record must be dynamically computed based on the user credentials. It should not be stored anywhere. If the database needs access to the records in plaintext format for the indexing purpose, it must have a role assigned to it and this role must be explicitly specified in the policies. Hence, after the database authenticates itself to the MET system and obtains its role credential, the database can compute the keys and use them to index different kinds of records. It should be noted that the computation cost incurred by decryption for indexing records is quite high. The solution to this problem is explained in section 4.3.2 where the database stores the IDs of the data owners in plaintext for rapid record indexing.

3.4.3.3 The Extension to the PBE system

By specifying the write action in conjunction with role-related rule in the policy and by using only the PBE system developed by Bagga and Molva fails to ensure that only the intended employee group can perform the write action to the document encrypted under the policy. According to the PBE scheme described in [39], the PBE system does not have the functionality for checking the compliance of a user's credentials with the policy. The compliance is automatically verified if the user's credentials can correctly decrypt the encrypted document. In the policy example given in section 3.4.2.2, both a doctor and a nurse can decrypt the encrypted document, make some changes to the document, encrypt the modified document and then send it

to the PBE system. The PBE system does not have a mechanism to distinguish between the document encrypted by the doctor and the one encrypted by the nurse. In [41], it is not clearly explained how to verify a user's compliance with a policy in order to enable the write action. If the MET system has a mechanism that verifies the compliance of a user with the policy based on his credentials (*e.g.* a signed role attribute assertion), then this mechanism functions in the same way as a traditional access control mechanism such as XACML. In this case the MET uses both the PBE system and an access control mechanism. This problem can be solved in an e-business environment where only the data owners (*i.e.* customers) should be able to change their private information stored on the company side. The solution is described in section 4.3.2.

3.5 Related Works Summary

This chapter provides summaries of the four related research works previously conducted in the areas of privacy enforcement architecture design and its complementary mechanisms. It also examines the problems of these related research works. The next chapter explains the theoretical contributions of this thesis research.

Chapter 4 Privacy Enforcement Architectures for an E-business Environment

This chapter clearly defines the user private data that needs to be protected in an e-business environment. This chapter then explains the Enhanced Privacy Enforcement Architecture Design using XML based on APEX (EAPEX) and its complementary mechanisms, as well as the theoretical design of the Architecture for Privacy Enforcement using PBE (APEP).

4.1 The User Private Data to be Protected in an E-business Environment

To various degrees, a variety of terms are used to specify data that identifies an individual in papers about privacy, privacy regulations and guidelines [24]. In the *European Union Directive 95/46/EC*, "an identifiable person" is defined as an individual who can be directly or indirectly identified by referring to an identification number or to a number of factors specifically related to his/her physical, physiological, mental, economic, cultural or social identity [31]. In Australia, the *Privacy and Personal Information Protection Act 1998* defines "personal information" as information and opinions about an identifiable person, including but not limited to written records about a person or a photograph or image of a person [32]. In Canada, under PIPEDA "personal information" is defined as information about an identifiable individual except the name, title or business address or telephone number of an employee of an organization [33]. In United States, different standards for identifiability of data are followed in different sectors. Terms such as "personally identifiable information (PII)" used in many other privacy documents are often not defined or are the cause for intensive debate [34].

According to the P3P Specification Working Group, the term "identified" data is defined as the information in a record or profile that can be tied to a particular person. The term "identifiable" data means any data that can be used reasonably by any other individual or by a data controller to identify a person. Therefore, in the P3P specification, "identified data" is a subset of "identifiable data" [24]. For example, a website server storing Internet Protocol (IP) addresses of visitors with their account information should consider the IP address to be "identified data," even though the IP addresses of most web users are dynamically allocated by their ISP over a long period of time. In the more common case where a website server stores visitors' IP addresses in web logs for maintenance and security purposes and uses aggregated IP addressing information or does not attempt to associate the IP address to a specific person or computer, IP addresses are not considered to be identified data even though law enforcement agents, for instance, can possibly use them to identify the individuals. The term "non-identifiable" data refers to anonymized information – that is, the data obtained by removing a part thereof in order to eliminate the connection to the individual's identity [24]. For instance, a web server might collect IP addresses but delete the last several digits of this information to ensure that a particular person or computer

cannot be identified. In the cases where no information is being collected at all, non-identifiable data can also be used [24].

In this thesis, two terms pertaining to the identifiability of data are defined. These are "identified" data and "identifiable" data. The term "identified" data is defined similarly to the one in P3P specification. It refers to the set of personal information that is commonly stored by websites/web service providers in their users' online accounts to facilitate service processing. For example, a web user's postal address is identified data in an e-business context because it is a common practice for e-businesses (*e.g.* Futureshop, Bestbuy) to store users' address information in their online accounts and to use it to facilitate online order processing. The term "identifiable" data refers to the difference between the set of "identifiable" data and the set of "identified" data defined in P3P 1.1. This difference is comprised of the personal information that is not stored in the user's online accounts and is not intended for identifying users but can be possibly used to do so. The identifiable data is the dynamic data defined in the base data schema of P3P (*e.g.* client IP address) [24]. The architectures proposed in this thesis focus on privacy protection enforcement over identified data only because identifiable data seems to have less business value than identified data and thus is less likely to be involved in privacy leak. Another reason is that most web users are not familiar with the web technologies related to identifiable data. Note that in the rest of the thesis, the expressions "identified data of users," "user private data" and "user private information" are used interchangeably.

The following sections are organized as follows. Section 4.2 describes the privacy enforcement architecture design called the Enhanced Architecture for Privacy Enforcement using XML (EAPEX), as well as its complementary mechanisms that make the involved technologies co-operate with each other. Section 4.2 describes the privacy enforcement architecture design based on the PBE system.

4.2 Enhanced Architecture for Privacy Enforcement using XML (EAPEX)

The APEX proposed by Barbieri in [1] provides a promising conceptual architecture design for enforcing privacy promises based on XML standards such as P3P, APPEL and XACML. Each architecture component is explained with possible technologies that can be used for implementation. However, this architecture design has many problems. Some of them are discussed but not handled by Barbieri [1], such as the purpose binding problem discussed in 3.1.3.1.3. The other problems found in the research for this thesis – like the misusing local privacy policy (LPP) problem discussed in section 3.1.3.2.1 – are not even mentioned in [1]. Furthermore, the mechanisms that connect the APEX components together are described in [1] only in terms of general ideas and possible technologies for implementation; the absence of implementation details for the mechanisms makes APEX less practical to follow.

Li and Yuan developed an automated mechanism that transforms XACML policies into P3P policies [10] while Mahmoudian implemented one that performs the transformation backwards [20]. However, since none of them offers a solution that addresses the semantic gaps between XACML and P3P, their transformation mechanisms are not practical from a general point of view.

The architecture design proposed in this section (called "Enhanced Architecture for Privacy Enforcement Using XML" or EAPEX in short) is based on APEX and on advances in privacy related XML standards like P3P 1.1 and XACML 2.0 with privacy policy profile. It makes use of the security framework model of XACML 2.0 which has been proven to be very practical and has been implemented by some well-known companies like Sun and IBM. The architecture design is explained in section 4.2.1.

The semantic gaps between P3P and XACML are handled by the Privacy Enforcement Extension of P3P 1.1 (P3PPEE) (discussed in section 4.2.2) which adds some new elements and defines the mandatory use of P3P optional elements like <recipient-description> to cover the semantic gaps.

Based on P3PPEE, section 4.2.3 proposes the mapping that precisely transforms privacy policies defined in P3P1.1 into the corresponding set of effectively functional XACML privacy control policies. This mapping can be used in the implementation of a mechanism that performs the transformation using XSLT in an automated fashion.

In order to reconcile a web user's privacy preferences and the privacy practices specified in a website/web service provider's P3P policies and in order to alleviate the burden of transforming the web user's privacy preferences regarding the P3P policies into privacy control policies (in XACML) on the server side, the privacy practice negotiation mechanism is proposed and explained in section 4.2.4.

4.2.1 Architecture Design

Privacy promises/practices are enforced using a privacy control mechanism which can be seen as a form of access control and which can be implemented using a traditional access control model such as the one included in XACML 2.0. The Enhanced Architecture for Privacy Enforcement using XML (EAPEX) proposed in this thesis describes how a privacy enforcement architecture can be structured and implemented using P3P and XACML for the policy specifications, using the security framework model of XACML 2.0 for managing the access to private data, and using APPEL for user privacy preferences. Figure 8 provides a design overview of EAPEX.

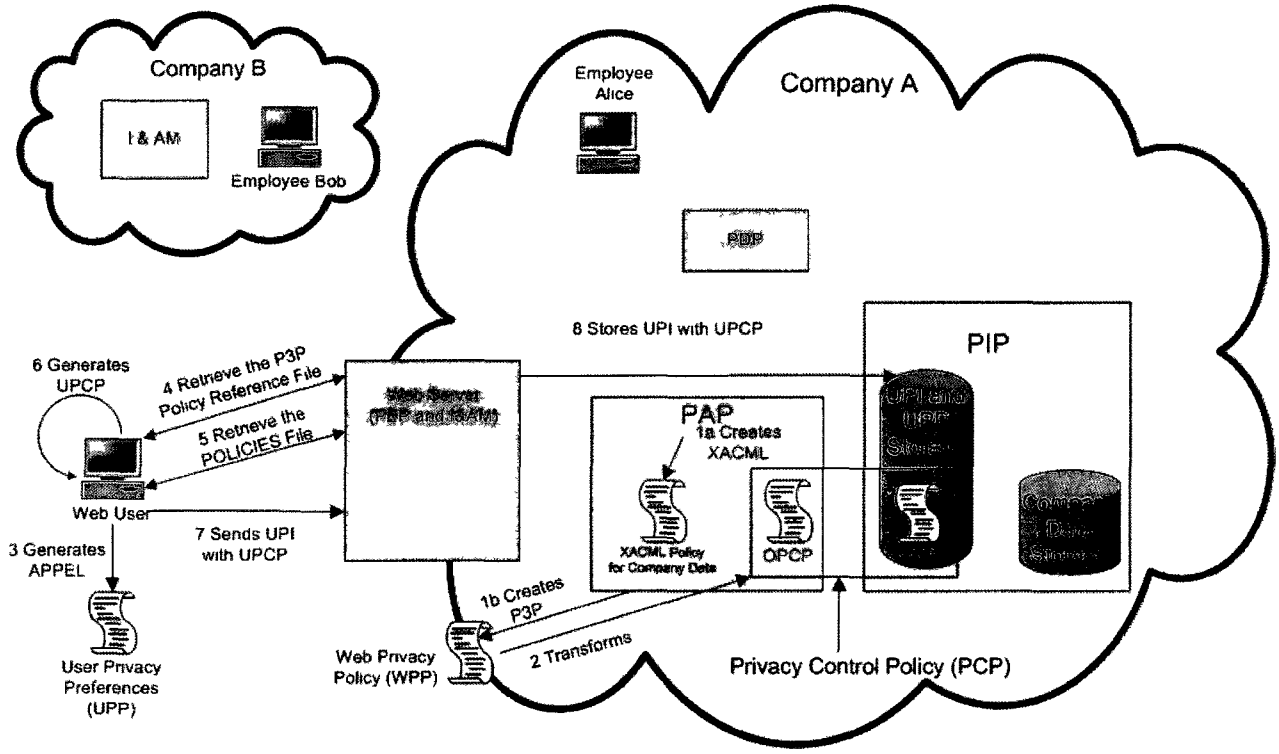


Figure 8 - Design Overview of EAPEX

This architecture design illustrates the major components of EAPEX. In this design, the Privacy Control Policy (PCP), the Web Privacy Policy (WPP) [1] and the User Privacy Preferences (UPPs) that are required for enforcing privacy are included to form the Privacy Policy Group (PPG). Among the three PPG components, the most important one is the set of Privacy Control Policies in which the negotiated privacy control practices are specified in XACML syntax. It is used by the Privacy Enforcement Model (PEM) to manage access to customers' private data in order to enforce the negotiated privacy practices. The major part of the PCP set that is derived from the general privacy promises specified in the P3P privacy policies of the company is called the Organizational Privacy Control Policy (OPCP) (section 4.2.1.1.1). It is obtained by using the Automated Privacy Policy Transformation mechanism (APPT) (section 4.2.3) to transform the WPP specified in P3P into the corresponding privacy control policies. The remaining part of the PCP set which represents customers' privacy preferences regarding the privacy promises of the company over their private data stored in their online accounts is called the User Privacy Control Policy (UPCP) (section 4.2.1.1.2). The UPCP is obtained via the EAPEX enabled user agent (section 4.2.4). The PCP is explained in detail in section 4.2.1.1. In the WPP, the privacy practices of the private data to be collected for the intended purposes to provide service are specified in P3P syntax. The WPP specified in P3P can be created using tools like JRC Policy Workbench [35]. Finally, in APPEL syntax the user's UPPs, which describe the preferred privacy promises among all the privacy promises specified in the WPP over the user's private

data, are automatically generated by the interaction between the user and the EAPEX enabled user agent (section 4.2.4). The set of XACML access control policies which are used by PCM to manage access to the other kinds of company data are included in the architecture design for illustrating the access control function of PCM only.

PPG is different from the various privacy policies included in PPA [1] as it takes user privacy preferences into serious consideration. In the Privacy Policy Architecture (PPA) [1], various kinds of privacy policies which are applied at different levels are defined in terms of concepts and possible technologies for implementation. They are the Inter-Organizational Privacy Policy (IOPP), the Organization Privacy Policy (OPP), the Local Privacy Policy (LPP) and the Web Privacy Policy (WPP) [1]. However, not all of these are applicable in an e-business setting. Since LPP specifies the rules to be enforced regarding transactions of private data on the local system only, it is not applicable to the transactions of private data between local system and web server. OPP and IOPP specify the rules regarding transactions of user private data within and between organizations and such rules can be specified in WPP by using P3P1.1 and P3PPEE. OPP and IOPP are thus removed from the architecture design. The UPPs remain in EAPEX and are used to control the web transaction since the users are the owners of the private data stored in their online accounts. Web users' privacy concerns are not addressed unless the privacy practices preferred by them are enforced.

The four major components of the security framework model of XACML are included to form the Privacy Control Model (PCM) in order to actually enforce the privacy control practices specified in the PCPs. These components are the Policy Administration Point (PAP), the Policy Decision Point (PDP), the Policy Information Point (PIP) and the Policy Enforcement Point (PEP). Note that the PEP is included in the web server which also functions as the website controller and the identity & access management system. Unlike the PAP defined in [27], the PAP is also responsible for transforming the WPP into the corresponding OPCP to manage access to users' private data in addition to creating XACML policies for other kinds of company data. The PIP is also different from the one defined in [27] as it is responsible for retrieving the appropriate UPCP in addition to the various applicable subject, environment and resource attributes needed by the PDP to make the XACML response regarding an access request to a particular user's private data. These major components of PCEM are explained in detail in the section 4.2.1.3.

The set-up operations of EAPEX shown in Figure 8 are explained as follows: 1a) the PAP creates XACML policies for managing access to the company data but not to the customers' private data; 1b) the PAP writes the privacy promises in P3P syntax; 2) the WPP is automatically transformed by PAP into the corresponding OPCP; 3) the user interacts with the APPEL enabled user agent to generate the UPP in APPEL; 4) upon visiting the website, the user agent sends an http request to retrieve the P3P policy reference file; 5) the user agent parses the P3P policy reference file and sends an http request to retrieve the POLICIES file that contains all the WPP policies covering the URL-space of the website; 6) once the user visits a web page that collects

user private information groups (UPIGs), the user agent parses the WPP covering the page and generates UPCP based on the UPP regarding the privacy promises in the WPP; 7) the user agent sends the User Private Information (UPI) and UPCP together to the website; and 8) the web server stores the UPI and UPCP in the UPI and UPP storage which is a database. It should be noted that it takes two round trips (*i.e.* operations 4 and 5) to retrieve the WPP.

4.2.1.1 Privacy Control Policy

In XACML, "named attribute" is defined as a specific instance of a characteristic of a subject, resource, action or environment. It is determined by the attribute name and type, the attribute holder's identity (*e.g.* subject, resource) and, optionally, the *identity of the issuing authority*. The attribute designator elements (*e.g.* <SubjectAttributeDesignator>) of XACML are used in a XACML rule to check the AttributeId, DataType and, most importantly, the *Issuer* of a particular attribute of the requester [27]. XACML is therefore capable of managing access requests by a subject whose attribute assertions are issued either by the company (*e.g.* Alice in company A) or by another organization (*e.g.* Bob in company B). In other words, controlling access to customers' private data within the organization and between the organizations can both be handled by using the PCPs specified in XACML. The OPCPs of the PCPs are derived from the WPPs. On the other hand, the UPCPs are generated by using the WPPs and the UPPs. Thus the UPP and WPP together have the same functionalities as the Inter-Organizational Privacy Policy (IOPP) [1] and the Organizational Privacy Policy (OPP) [1] in the original APEX.

4.2.1.1.1 Organizational Privacy Control Policy

The Organizational Privacy Control Policies (OPCPs) specify the privacy control practices of the company regarding customers' private data in the XACML syntax. The privacy practices are firstly published as Web Privacy Policies in P3P with P3PPEE (section 4.2.2). By using P3P with P3PPEE, the company can precisely express its privacy practices without the problem of being ambiguous or misleading. For example, the following policy segment in P3P with P3PPEE represents the privacy promise – that is, the "user's email will be used by only the marketing advisors within company A for marketing purposes." The key information embedded is in bold.

Listing 4.2.1.1.1a The Privacy Promise Example Specified in P3P

```
<STATEMENT>
 <EXTENSION><p3p11:STATEMENT-GROUP id="account" /></EXTENSION>
 <DATA-GROUP>
  <DATA ref="#user.home-info.online.email">
 </DATA-GROUP>
 <PURPOSE>
  <EXTENSION>
   <p3ppee:PURPOSE-ROLES>
```

```
      < p3ppee:marketing>
       < p3ppee:role>MarketingAdvisor</ p3ppee:role>
      </ p3ppee:marketing>
     </p3ppee:PURPOSE-ROLES>
    </EXTENSION>
   </PURPOSE>
   <RECIPIENT>
    <ours><recipient-description>CompanyA</recipient-description></ours>
   </RECIPIENT>
</STATEMENT>
```

Then these precisely described privacy promises are translated into XACML syntax to form the OPCPs. The code segment below represents the corresponding privacy control practice of the above privacy promise example in the resulting OPCP.

Listing 4.2.1.1.1b The Privacy Promise Example Specified in XACML

```
<Rule Effect="Permit">
 <Target>
  <Subjects>
<!-- target subject who has a role attribute issued by "companyA" with value " MarketingAdvisor " -->
    <Subject>
     <SubjectMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
      <AttributeValue> MarketingAdvisor </AttributeValue>
      <SubjectAttributeDesignator    AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:role"
DataType="http://www.w3.org/2001/XMLSchema#string" Issuer="companyA"/>
     </SubjectMatch>
    </Subject>
  </Subjects>
  <Resources>
<!-- target resource which is the email elements in customer online account -->
    <Resource>
     <ResourceMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
       urn:example:companyA:schemas:record
      </AttributeValue>
     <ResourceAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:resource:target-namespace"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
     </ResourceMatch>
     <ResourceMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:xpath-node-match">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
       /CustomerRecord/account/email
      </AttributeValue>
      <ResourceAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:resource:xpath"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
     </ResourceMatch>
    </Resource>
  </Resources>
<Actions>
<!-- the target subject can only read the target resource for marketing purpose only -->
    <Action>
     <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
      read
      </AttributeValue>
      <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
DataType="http://www.w3.org/2001/XMLSchema#string" Issuer="companyA"/>
```

```
      </ActionMatch>
      <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
       <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
       marketing
       </AttributeValue>
       <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:2.0:action:purpose"
DataType="http://www.w3.org/2001/XMLSchema#string" Issuer="companyA"/>
      </ActionMatch>
     </Action>
   </Actions>
  </Target>
</Rule>
```

The transformation of the WPPs specified in P3P with P3PPEE into the corresponding OPCPs can be conducted in an automated fashion by using either XSLT or a programming language like Java with the Document Object Model (DOM) by W3C [36]. Since the XSLT stylesheet module is easier for an external auditor to use in verifying the consistency between the P3P policies and the resulting XACML policies, XSLT is the technology used by the e-business in EAPEX. The complete mapping pattern for transforming P3P privacy policies into the corresponding OPCPs is described in section 4.2.3.

4.2.1.1.2 User's Privacy Control Policy

The User's Privacy Control Policies (UPCPs) specify the customers' privacy preferences relating to the general privacy promises of the company with regard to the UPI stored in their online accounts. Firstly a customer interacts with an APPEL enabled user agent such as an internet browser to generate his privacy preferences in APPEL syntax. For example, a user can inform the user agent that he does not want any website to use his email address for marketing purposes by checking the appropriate checkbox. Then the user agent generates an APPEL preference which is illustrated in the following code segment.

Listing 4.2.1.1.2a The Privacy Preference Example Specified in APPEL

```
<appel:RULESET>
...
 <appel:RULE behavior="limited" description="Service collects email for marketing purpose">
  <p3p:POICY>
   <p3p:STAEMENT>
    <p3p:DATA-GROUP>
     <p3p:DATA>
      <p3p:DATA ref="#user.home-info.online.email"/>
     </p3p:DATA>
    </p3P:DATA-GROUP>
    <p3p:PURPOSE>
     <p3p:EXTENSION>
      <p3ppee:PURPOSE-ROLES>
       <marketing/>
      </ p3ppee:PURPOSE-ROLES>
```

```
      </p3p:EXTENSION>
     </p3P:PURPOSE>
    </p3p:STATEMENT>
  </p3p:POLICY>
 </appel:RULE>
...
</appel:RULESET>
```

The behaviour output used in this APPEL rule is "limited" in the sense that, if the website did require user to submit his email address for various purposes including <marketing/> purpose, the user agent should submit the email address but remove <marketing/> purpose from the allowable purposes (*i.e.* limits the allowable purposes). This is achieved by the user agent generating a UPCP rule that denies the non-preferred privacy practice and submitting the email address with the resulting UPCP rule together to the web controller.

For example, a user is trying to create an online account on the website of Company A. He decides to use "charlie" as the username. The registration page requires the user to submit his email address. The P3P policy covering the page includes <marketing/> purpose with the contained employee role element <p3ppee:role> **MarketingAdvisor**</ p3ppee:role> (listing 4.2.1.1.1a). The user agent will generate a UPCP rule in XACML syntax which denies any access by employees of the company who have a **MarketingAdvisor** role to the user's email address for the purpose of marketing based on the user's privacy preference related to the privacy promise specified in P3P with P3PPEE (listing 4.2.1.1.2a). The generated UPCP rule is shown in listing 4.2.1.1.2b.

Listing 4.2.1.1.2b The Generated User's Privacy Control Policy Example

```
<Rule Effect="Deny">
 <Target>
  <Subjects>
<!-- target subject who has a role attribute issued by "companyA" with value "MarketingAdvisor" -->
    <Subject>
     <SubjectMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
      <AttributeValue> MarketingAdvisor</AttributeValue>
      <SubjectAttributeDesignator    AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:role"
DataType="http://www.w3.org/2001/XMLSchema#string" Issuer="companyA"/>
     </SubjectMatch>
    </Subject>
  </Subjects>
  <Resources>
<!-- target resource which is the email elements in customer online account -->
    <Resource>
     <ResourceMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
       urn:example:companyA:schemas:record
      </AttributeValue>
     <ResourceAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:resource:target-namespace"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
     </ResourceMatch>
```

```
      <ResourceMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:xpath-node-match">
       <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
        /CustomerRecord/account/email
       </AttributeValue>
       <ResourceAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:resource:xpath"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
      </ResourceMatch>
     </Resource>
    </Resources>
<Actions>
<!-- the target subject can only read the target resource for marketing purpose only -->
     <Action>
      <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
       <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
       read
       </AttributeValue>
       <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
DataType="http://www.w3.org/2001/XMLSchema#string" Issuer="companyA"/>
      </ActionMatch>
      <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
       <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
       marketing
       </AttributeValue>
       <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:2.0:action:purpose"
DataType="http://www.w3.org/2001/XMLSchema#string" Issuer="companyA"/>
      </ActionMatch>
     </Action>
    </Actions>
  </Target>
<!-- check if the access requested is particularly to charlie's online account -->
  <Condition>
   <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-equal" >
    <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string" >
     "charlie"
    </ AttributeValue>
    <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-one-and-only">
     <AttributeSelector RequestContextPath="//xacml-context:Resource/xacml-context:ResourceContent/
CustomerRecord/account/id/text()" DataType="http://www.w3.org/2001/XMLSchema#string"/>
    </Apply>
   </Apply>
  </Condition>
</Rule>
```

The P3P with P3PPEE allows the website to describe its privacy promises in varying degrees of detail. If the P3P policy includes only the <marketing> purpose but without any embedded employee role, the generated UPCP rule will be similar to the one in listing 4.2.1.1.2b except that the <Subjects> element in the <Target> element is removed. It then denies any access to charlie's email address for the purpose of marketing by the employees with any role in Company A.

The mapping patterns for generating UPCP rules based on the user's UPPs relating to the privacy promises of the company with regard to the UPI stored in his/her online accounts are explained in detail in section 4.2.4.

4.2.1.2 The Privacy Enforcement Model

The Privacy Enforcement Model (PEM) consists of four components: Privacy Administration Point (PAP), Policy Decision Point (PDP), Policy Information Point (PIP), and Policy Enforcement Point (PEP). These components are originally introduced in the security framework model of XACML to manage access. In PEM, they are also used to enforce privacy control practices. The operations performed by the PEM to enforce the negotiated privacy control practices are illustrated in Figure 9. The operations are as follows: 1a) an employee named Alice signs on to the Identity and Access Management system (I&AM) of the Company A; 1b) a worker named Bob signs on to the I&AM system of company B; 1c) a customer of company A named Charlie signs on to the I&AM system of company A; 2) the subject (2a: Alice, 2b: Bob, or 2c: Charlie) sends a request in the native format (*e.g.* in SOAP format) for access to a particular type of Charlie's UPI (*e.g.* his postal address) to the PEP of the website of Company A; 3) the PEP forms a XACML request based on the request in the native format and sends it to the PDP; 4) the PDP acquires the applicable OPCP from the PAP (*e.g.* the OPCP used to manage access to customers' postal address); 5) the PDP obtains the UPCP rule stored with the requested UPI (*e.g.* the UPCP used to manage access to Charlie's postal address) in addition to the requested UPI of Charlie, as well as the relevant subject, environment and resource attributes; 6) the PDP makes an authorization decision based on the OPCP, the UPCP rule, the UPI and the attributes and then sends it in a XACML response to the PEP; and 7) if the access is permitted, the PEP permits the access to Charlie's postal address. Otherwise it denies the access.

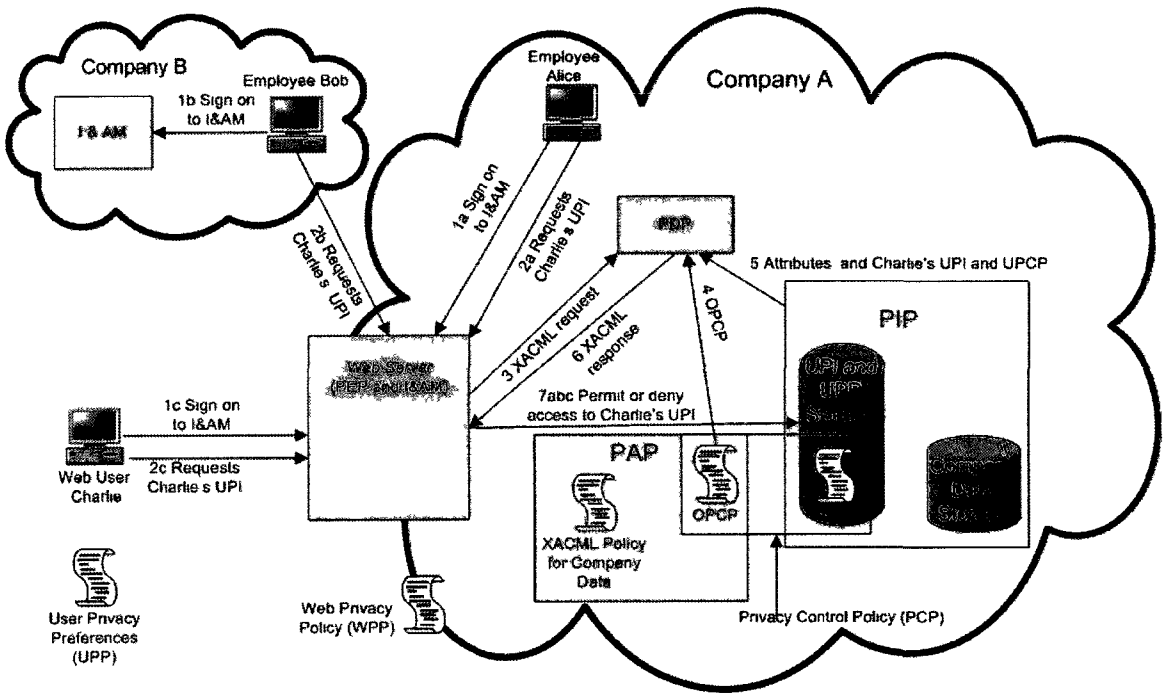


Figure 9– EAPEX Access Request Processing of PCM

4.2.1.2.1 Privacy Administration Point (PAP)

The PAP of the PCM is responsible for creating the access control policies which are used to manage access to the company's internal corporate data. In addition, the PAP is also responsible for creating the Web Privacy Policies (WPPs) in P3P with P3PPEE and then transforming them into the corresponding Organizational Privacy Control Policies (OPCPs). Note that additional transformation is required to keep the OPCPs consistent with the WPPs when the WPPs are updated or modified.

4.2.1.2.2 Privacy Decision Point (PDP)

When a subject sends a request for accessing a particular piece of UPI to the PCM, the PDP evaluates the target elements of the OPCPs to find the applicable ones. Then it evaluates the applicable OPCPs together with the relevant UPCP rule (the UPCP rule which is stored with that particular piece of UPI) to make a XACML response.

4.2.1.2.3 Policy Information Point (PIP)

The Policy Information Point is responsible for retrieving the requested UPI and the associated UPCP rule in addition to the various subject, environment and resource attributes required by the PDP to make the authorization decision regarding an access request to a particular piece of UPI.

4.2.1.2.4 Policy Enforcement Point (PEP)

Unlike APEX whereby different Privacy Enforcement Engines (PEE) are deployed to enforce privacy control practices regarding transactions of UPI at different levels, a single Policy Enforcement Point (PEP) is used in the PCM of EAPEX to enforce the negotiated privacy control practices regarding access to the UPI stored on the company side. A single PEP is sufficient since the PCP is capable of managing both local access (*i.e.* access request issued within the company) and remote access (*i.e.* access request issued outside of the company) to the UPI as explained in section 4.2.1.1.

4.2.2 The Privacy Enforcement Extension of P3P 1.1

XACML [27] is an especially powerful language for specifying concrete access control policies that can be used by the included security framework model to achieve access control. With the privacy policy profile extension [30], effective privacy control practices which are followed to manage access to the UPI can be specified in XACML policies. Meanwhile, P3P is established to enable websites and companies to publish their private promises/practices in machine readable format thus alleviating the ambiguity and the problem of misleading privacy policies written in

human readable format. P3P policies are mainly used by the P3P enabled user agents to help web users better understand the privacy promises and make appropriate decisions while using online services. In general, privacy policies specified in P3P have insufficient semantics to be transformed into the corresponding effective privacy control policies in XACML.

There are major semantic gaps between P3P and XACML. For example, the base data schema of P3P only specifies the most common types of User Private Information (UPI) that can be collected by a website, while the <ResourceMatch> element of XACML requires a specific identifier (*i.e.* the node name for generating the XPath expression) for each type of the collected UPI in order to locate the data. Thus, if a website collects a particular type of UPI (*e.g.* credit card information) which can only be denoted by using variable category elements in P3P such as <miscdata>, the XPath expression for locating this kind of UPI used in the <ResourceMatch> element cannot be precisely generated (solution explained in 4.2.2.1). In another example of the semantic gap between P3P and XACML, according to Yee, the employee role check of the requesting subject needs to be performed in addition to the purpose check of the requested action in order to enforce the privacy practices regarding the UPI [37]. However, the syntax for expressing roles and for binding between roles and the associated purposes is provided only in XACML but not in P3P (solution explained in 4.2.2.2). There are also some minor semantic gaps between P3P and XACML. In this thesis, P3PPEE is developed to extend P3P 1.1 so that those semantic gaps can be covered. Note that, unless the gaps mentioned above are properly covered, any automated transformation mechanism that transforms P3P policies into the corresponding effective XACML policies or vice versa is impractical (as explained in sections 3.2.3 and 3.3.4).

In this section, the Privacy Enforcement Extension of P3P 1.1 (P3PPEE) is proposed in order to clarify the ambiguities of P3P 1.1 and thereby cover the semantic gaps between P3P and XACML that cause mapping problems while transforming P3P 1.1 policies into the corresponding XACML policies. It extends the base data schema of P3P 1.1 and adds the new element <PURPOSE-ROLES> through the extension mechanism of P3P 1.1 to cover the major semantic gaps. In addition, it defines the mandatory use of some optional P3P 1.1 elements and uses the extension mechanism of P3P 1.1 to cover the minor semantic gaps. With P3PPEE in place, the transformation of P3P privacy policies into the corresponding Organizational Privacy Control Policies (OPCPs) becomes achievable. The XML definition code of P3PPEE is shown in appendix 1.

4.2.2.1 Extending Base Data Schema of P3P

In a P3P policy, data types (<DATA/>) must be included in the <DATA-GROUP> of each <STATEMENT> element to specify the types of dynamic data and user identified data to which the described privacy promises apply. The base data schema of P3P 1.1 is the default hierarchical set of data types of increasing granularity that describes the set of data types which can be used

within <DATA-GROUP> elements [24]. A major disadvantage of the base data schema is that various kinds of customer identified data which are commonly collected by e-commerce websites cannot be directly specified with reference to the data types defined in the base data schema. For example, credit card information is commonly collected by e-business websites as online payment information to process customer online orders. Credit card information is certainly a piece of user identified data because it contains a user's real name and a unique credit card number associated with that name. The credit card information should be able to refer to a detailed data type of user data type. However, in P3P 1.1, if the base data schema is used, the credit card information can only be denoted by the <miscdata> data type associated with <purchase/>. The <miscdata> element is as follows

```
<miscdata>
 <CATEGORIES>
  <purchase/>
 </CATEGORIES>
</miscdata>
```

Although variable-category data elements of P3P such as <miscdata> with the attached category tags can be used to describe any type of identified data other than those defined in the base data schema, the ambiguity within <miscdata> itself can cause serious problems. According to P3P specification 1.1, the <miscdata> element shown above can also be used to refer to the actual purchase order (*i.e.* purchase order items) and thus cause ambiguity. Furthermore, the ambiguity problem becomes even worse after the <miscdata> is transformed into XACML because the <CATEGORIES> element which helps to alleviate the ambiguity of the <miscdata> element is dropped. The following XACML fragment is derived by mapping the <miscdata> element using the ATPX proposed in [20].

```
<xacml:ResourceMatch>
 <xacml:AttributeValue>dynamic.miscdata</xacml:AttributeValue>
 <xacml:ResourceAttributeDesignator AttributeId="resource:resource-id" DataType="string"/>
</xacml:ResourceMatch>
```

The resulting resource-id value "dynamic.miscadata" is used to check the identity of the resource being requested. But the id value is very confusing since the attached <CATEGORIES> element cannot be transformed into XACML and is thus dropped during transformation using [20]. Hence it possibly represents any kind of the collected user identified data whose type is not clearly defined in the base data schema.

In order to eliminate the ambiguity caused by using the variable-category data elements of the base data schema, the data type <user> of the base data schema of P3P 1.1 is replaced by the extended one which is shown as the follows:

```
<complexType name="datadefComplexType">
```

```
<all>
<element minOccurs="0" name="dynamic" type="p3p11bds:dynamicComplexType" />
        <!-- ********** Substitue ********** -->
<element minOccurs="0" name="user" type="p3ppee:userComplexType" />
<element minOccurs="0" name="thirdparty"    type="p3p11bds:thirdpartyComplexType" />
<element minOccurs="0" name="business" type="p3p11bds:businessComplexType" />
</all>
<attribute type="p3p:yes_no" default="no" use="optional" name="optional" />
</complexType>
```

The extended data type <user> is derived by extending the original one with an additional allowable contained element <payment>:

```
<complexType name="userComplexType">
 <complexContent>
  <extension base="p3p11bds:userComplexType">
   <all>
    <element minOccurs="0" name="payment" type="p3ppee:paymentComplexType">
     <annotation>
      <appinfo>
       <CATEGORIES xmlns="http://www.w3.org/2002/01/P3Pv1">
        <financial />
       </CATEGORIES>
      </appinfo>
      <documentation>User's Payment Information</documentation>
     </annotation>
    </element>
   </all>
  </extension>
 </complexContent>
</complexType>
```

The <payment> element is then used in turn to define the child element <creditcard> and its contained elements. The new hierarchy of the user data type is shown as follows:

```
<user>
 ...
 <payment>
  <creditcard/>
  <cctype/>
  <ccnum/>
  <holdername/>
  <expiry/>
  <securitycode/>
 </payment>
</user>
```

Similarly, the base data schema can be extended to include any kind of user identified data other than those defined therein. Possible examples include a medical card, passport, driver's licence and so forth. The complete extension code for including credit card information in the base data schema is given in appendix 1.

4.2.2.2 Adding the <PURPOSE-ROLES> Element through the Optional <EXTENSION> of <PURPOSE>

As the XACML code example provided by Yee in [37] demonstrates, both the employee role of the requesting subject and the action purpose of the action requested need to be checked to enforce privacy control practices regarding customers' private data. However, the action purposes and the associated employee roles cannot be expressed using P3P 1.1. In P3PPEE, the new element (*i.e.* <PURPOSE-ROLES>) is defined to enable P3P to express such semantics.

4.2.2.2.1 Conciliating <PPURPOSE> and <PURPOSE>

In P3P 1.1, the new element <PPURPOSE> which is added through <EXTENSION> is used to specify the primary purpose for information collection in addition to the <PURPOSE> element defined in P3P 1.0. It is intended to expand upon the <current/> tag to provide a more detailed explanation of data usage [24]. However, the relationships between the expanded <current/> purpose tag and the other purpose tags are not well explained in the P3P specification 1.1. Moreover, examples of using <PPURPOSE> element are not provided either [24]. There are overlaps between the expanded <current/> tag and the other purpose tags. For example, a website page prompts a user to submit his/her telephone number in order to inform him/her about the current promotion. In this case, either the <current/> tag in the <PURPOSE> element with the <marketing/> tag specified in the <PPURPOSE> element or the <telemarketing/> tag specified in the <PURPOSE> element can be included to denote the purpose of collecting the user's telephone number. In order to remove the redundancies caused by the overlaps, a new element called <PURPOSE-ROLES> is defined in the Privacy Enforcement Extension of P3P. Its purpose tags are derived by replacing the <current/> tag element with the ones introduced in the <PPURPOSE> element and then eliminating the duplicated original purpose value tags. The <PURPOSE-ROLES> element can be included in P3P policies through <EXTENSION> elements of the <PURPOSE> element. The tags of the <PURPOSE> element that need to be removed are listed as follows:

<current/>: This means that the information collected is used by the service provider to complete the activity requested by a user. It is the purpose tag which the <PPURPOSE> intends to expand. Thus it is not included in the <PURPOSE-ROLES> element.

<contact/>: This indicates that the purpose of information collection is to contact the individual through a communications channel other than voice telephone for the promotion of a product or service. It can be substituted with <marketing/>.

<telemarketing/>: This means that the information collected is used to contact the individual via a voice telephone call for promoting a product or service. Similarly, <marketing/> can be used instead.

<other-purpose>string</other-purpose>: This is the variable purpose tag. It specifies the ways of using collected information that are not captured by the other <PURPOSE> values. This purpose value is removed from <PURPOSE-ROLES>. The reason is that, in the context of a particular e-business, if there is a purpose that cannot be denoted by any tag of the <PURPOSE-ROLES> element, then it can be easily included by modifying the definition of <PURPOSE-ROLES>. So there is no need to keep this variable purpose tag.

The XML schema definition of the element <PURPOSE-ROLES> is shown as follows:

Definition of <PURPOSE-ROLES>:

```
<element name="PURPOSE-ROLES">
 <complexType>
  <sequence>
        <choice maxOccurs="unbounded">
         <element name="account" type="p3ppee:purpose-role-value"/>
         ...
    </choice>
   </sequence>
 </complexType>
</element>
```

4.2.2.2.2 Binding Employee Roles to Purposes

In addition to the purpose value elements of the <PURPOSE-ROLES> element derived by the procedures described above, the <PURPOSE-**ROLES**> element defined in P3PPEE allows organizational employee role tags to be associated in a contained purpose value element, in order to indicate the bindings between the purposes and the roles. So "ROLES" is included as part of the element name.

It is necessary to allow the e-business to explicitly express the purpose-role bindings; otherwise the resulting action purpose check of the OPCP generated by the automated transformation mechanism can be bypassed by the company intentionally. For example, a website includes the following P3P code segment for the collection of users' email addresses to inform users that their email addresses will be collected for creating and updating their online accounts only.

```
<PURPOSE-ROLES>
 <account/>
</PURPOSE-ROLES>
```

The corresponding XACML code is shown as follows:

```
<ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
      account
      </AttributeValue>
      <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:2.0:action:purpose"
DataType="http://www.w3.org/2001/XMLSchema#string" />
```

```
    </ActionMatch>
</ActionMatch>
```

The XACML code checks the purpose of access to users' private data only. Since the code for checking the purpose-role binding is missing, it is possible that the company allows its marketing employees to access the users' emails by issuing to them the attribute assertion "action:purpose=account".

With the inclusion of purpose-role bindings, the company has to include the following P3P segment in order to do so (such an abnormal purpose-role binding can be easily detected by the auditor by validating the <PURPOSE-ROLES> element):

```
<PURPOSE-ROLES>
 <account>
   <MarketingAdvisor/>
   ...
 </account>
</PURPOSE-ROLES>
```

The purpose-role bindings of a particular e-business can be derived from its structure. Enterprises consist of several major divisions and a certain set of functions is carried out by each major division. A typical large enterprise usually has following divisions: research, development (sometimes a combined with research to form a division called R&D), legal, finance, sales, marketing, customer service and human resources. Among these, only the functions of a few divisions use customers' private data. In other words, only the employees with certain roles and only particular systems such as the web server should be allowed to access customers' private data in order to carry out their work. For example, in a typical e-business, the web server of its website should have access to customers' private data in order to help customers to maintain their online accounts and to provide the services requested by customers. Therefore the purpose-role bindings can be generated by using the roles and the purposes of the work associated with them. Note that the cardinality between purpose tags and role tags is one-to-many. This is why 'S' is included in the element name (<PURPOSE-ROLES>). For instance, the web server of an e-business collects a customer's postal code address to update his/her online account, while a customer service representative of the e-business can also use the information for the same purpose.

The following XML schema code fragments are the definition of the <role> element that can be bound to the <PURPOSE-ROLES> element values:

```
<complexType name="purpose-role-value">
        <sequence>
                <element ref="p3ppee:role" minOccurs="1" maxOccurs="unbounded"/>
        </sequence>
        <attribute name='required' use='optional' type='p3p:required-value'/>
```

```
</complexType>

<element name="role">
        <simpleType>
                <restriction base="string">
                        <enumeration value="MarketingAdvisor"/>
                        ...
      </restriction>
        </simpleType>
</element>
```

4.2.2.3 Mandatory Use of Optional P3P Elements and of the Extension Mechanism

4.2.2.3.1 <STATEMENT-GROUP-DEF> and <STATEMENT-GROUP>

In P3P, use of the <STATEMENT-GROUP-DEF> element is optional. It is used to define an identifier and, optionally, properties that can be applied to a group of <STATEMENT> elements [24]. In P3PPEE, the <STATMENT-GROUP-DEF> elements are mandatory and are used to indicate the User Private Information Groups (UPIGs) used within the e-business. The names of the UPIGs are indicated by the ID attributes of the <STATEMENT-GROUP-DEF> elements. UPIG is a new term that is based on the intuitive idea that related kinds of User Private Information (UPI) are often collected and used together for a particular purpose and are also often stored together, thus forming a UPIG. Hence, in a website where EAPEX is deployed, the related kinds of UPI in the same UPIG are stored under the same parent element in the XML formatted customer record and in same table in UPI and UPP storage (*i.e.* the database). In other words, the schemas of the tables (or the schema of the XML formatted customer records) for storing UPI define UPIGs. For example, the street number and name, city, province, postal code and country of a customer are collected and used together by an e-business for marketing and feedback purposes. These kinds of UPI are stored in the table named "address". Hence, a UPIG called "address" is defined using the <STATEMENT-GROUP-DEF> element to group the related kinds of the UPI together. The name of the UPIG is the same as the name of the table that stores the user's postal address information and the name of the parent element in the XML formatted customer record that contains the address detail elements (*e.g.* <street>, <city>, etc.).

Unlike in P3P where <STATEMENT-GROUP> is optional and is used to associate a <STATEMENT> element to a statement group defined by using <STATEMENT-GROUP-DEF> element [24], this element is mandatory for every UPI-collection-related <STATEMENT> element and is used to associate each <STATEMENT> element with a UPIG. For instance, user ID, password and email are commonly used together to create an online account, so a UPIG called "account" is defined to aggregate these kinds of UPI. After the UPIG is indicated using a <STATEMENT-GROUP-DEF> element, a <STATEMENT>, which states the data collection of this UPIG, can refer to it through the contained <STATEMENT-GROUP> element. Later, when transforming the <STATEMENT> element into the corresponding XACML rule, the name of the

indicated UPIG will be used to generate the XPath expression (*i.e.* /account) which locates the UPI stored in the XML formatted customer record. Sometimes, only a particular kind of UPI of a UPIG is needed for a specific purpose. For instance, the email type of UPI of the UPIG "account" is also used for a <marketing/> purpose in addition to the <account/> purpose. In this case, the <STATEMENT> element which specifies the marketing usage of the user's email also needs to indicate the UPIG "account". When transforming this statement into the corresponding XACML rule, the UPIG name and the UPI type name are used together to generate the XPath expression (*i.e.* /account/email) which locates users' email information in the XML formatted customer records. Therefore, with P3PPEE, every <STATEMENT> element regarding UPI collection – regardless of its <DATA-GROUP> – includes all the UPI types of a UPIG or only some of them and must refer to the UPIG through its <STATEMENT-GROUP> element.

4.2.2.3.2 <recipient-description>

According to P3P 1.1, each of the recipient category tags (*e.g.* <ours>, <same>, etc.) can optionally contain one or more <recipient-description> tags which each include the description of a recipient. In P3PPEE, each of the recipient category tags contained in a statement regarding data collection of a UPI must contain <recipient-description> tags. These tags are used to specify the identities of the organization recipients included in the recipient category. By doing this, the EAPEX enabled websites have to explicitly expose all the organization recipients of the UPI collected. When transforming a UPI collection related statement into the corresponding XACML rule, the identities of the organization recipients are used to generate the XACML code that checks the issuer of subject attributes. Note that, in situations where P3P privacy policies and P3P enabled user agents are already in place, it is very unlikely that a web user would allow the company to share his/her identified data with other companies and organizations that possibly follow completely different privacy practices. Therefore, only the <ours> and <same> category elements should be considered as the acceptable elements in the <RECIPIENT> element.

4.2.2.3.3 The Extension Element of <RETENTION>

In P3P 1.1, the time period of the User Private Information (UPI) retention is described in the <RETENTION> element. If the <RETENTION> element contains the <no-retention/> element, the time period is zero. If it contains the <indefinitely/> element, then the time period is considered to be infinity (the worst case) in this thesis. If it includes either the <stated-purpose/>, <legal-requirement/> or <business-practices> element, the time period of UPI retention is provided in a supplementary resource. To eliminate the need for the supplementary resource, the use of the extension element of the <RETENTION> element to specify the exact time period of UPI retention is required in P3PPEE for every UPI collection related statement. The following example shows how a time period of two years is specified in the <EXTENSION> element of the <RETENTION> element:

```
<p3p:RETENTION>
 <EXTENSION  xmlns:xf="http://www.w3.org/TR/2002/WD-xquery-operators-20020816/#" >
  <xf:dt-yearMonthDuration>
  P2Y
  </xf:dt-yearMonthDuration>
 </EXTENSION>
 <business-practitices/>
</p3p:RETENSION>
```

A time period of retention specified in this way can be directly used to form the XACML code which checks if the retention of UPI is expired during the transformation of P3P with P3PPEE into XACML.

4.2.3 Mapping Patterns for the Transformation of P3P with P3PPEE into XACML

The effectiveness of the Privacy Control Model (PCM) depends on the effectiveness of the Privacy Control Policy (PCP) which, in turn, depends on the effectiveness and accuracy of the mapping of P3P policies to XACML policies. With the Privacy Enforcement Extension of P3P (P3PPEE) explained in the previous section which covers the semantic gaps between P3P and XACML, an accurate mapping which generates effective Organizational Privacy Control Policies (OPCPs) based on P3P policies is proposed.

In this section, the mapping patterns for transforming policies written in P3P with P3PPEE into the corresponding XACML policies (OPCPs) are explained in detail. These mapping patterns can be specified in a XSLT sheet and be used by a XSLT engine to perform the transformation automatically. Note that the <ACCESS> element of P3P seems to be very ambiguous according to the P3P specification, especially the containable <other-ident/> element. Neither the precise definition nor clear examples of using it are given in P3P specifications 1.0 [6] or 1.1 [24]. As explained earlier in the beginning of section 4, "identified data" refers to the set of personal information that is commonly stored by e-businesses in their users' online accounts to facilitate service processing. Examples include email, postal address, telephone number, online payment and so forth. In the most common case, e-commerce websites (like those of Futureshop and Bestbuy) allow customers to access all of the identified information stored in their online accounts in order that they be able to modify or update their information when needed. Thus, only the <ACCESS> element which contains <nonident/>, <all/>, or <none/> is considered relevant to the mapping discussed in this section. Also note that the company needs to apply the mapping to generate up-to-date OPCPs whenever it updates or modifies its WPPs. In the rest of this section, P3P is used to denote P3P 1.1 with P3PPEE for simplicity.

4.2.3.1 An Example of the P3P Policy Segment

The following P3P segment, taken from the P3P policy, covers the registration page of Automart (section 5.1). It will be used to illustrate the P3P patterns used by the mapping to generate the corresponding XACML code in section 4.2.3.3. The privacy promises described by the contained <p3p:STATEMENT> element are as follows:

> Automart collects customers' email address for our email flyer server, to send email flyers upon user consent. We retain the collected email addresses for two years. We share customer email addresses with company, Automodel, which uses them under equitable privacy practices. Users can access their email address and the time of account creation.

```
<POLICY discuri="/privacy_policy.jsp" name="RegisterPage" opturi="/privacy_policy.jsp">
        <EXTENSION>
                <p3p11:STAEMENT-GROUP-DEF id="account" consent = "mixed" short-description="Account
Registration Necessaries"/>
                ...
        </EXTENSION>
    <ENTITY>
     ...
    </ENTITY>
    <ACCESS>
      <all/>
    </ACCESS>
    <DISPUTES-GROUP>
     ...
    </DISPUTES-GROUP>
    <STATEMENT>
      <EXTENSION>
              <p3p11:STATEMENT-GROUP id="account" />
      </EXTENSION>
      <CONSEQUENCE>We collect your email address to create your online account and process your order  You can
choose to receive flyers by email</CONSEQUENCE>
      <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
      <DATA ref="#user.home-info.online.email"/>
      <DATA ref="#dynamic.clickstream.timestamp"/>
     </DATA-GROUP>
     <EXTENSION>
      <DATA-GROUP xmlns="http://www.w3.org/2004/02/P3Pv11">
       <p3ppee:datatype>
        <p3ppee:user>
         <p3ppee:home-info>
          <p3ppee:online>
           <p3ppee:email/>
          </p3ppee:online>
         </p3ppee:home-info>
        </p3ppee:user>
        <p3ppee:dynamic>
         <p3ppee:clickstream>
          <p3ppee:timestamp>
         </p3ppee:clickstream>
        </p3ppee:dynamic>
       </p3ppee:datatype>
      </DATA-GROUP>
     </EXTENSION>
     <PURPOSE>
```

```
        <EXTENSION>
          <PPURPOSE xmlns="http://www.w3.org/2004/02/P3Pv11">
            <marketing/>
          </PPURPOSE>
        </EXTENSION>
        <contact required="opt-out"/>
        <EXTENSION>
                <p3ppee:PURPOSE-ROLES>
                        <p3ppcee:marketing>
                                <p3ppcee:role>EmailFlyerServer</p3ppcee:role>
                        </p3ppcee:marketing>
                </p3ppee:PURPOSE-ROLES>
        </EXTENSION>
      </PURPOSE>
      <RECIPIENT>
        <ours>
                <recipient-description>www.automart.com</recipient-description>
        </ours>
        <same required="opt-out">
                <recipient-description>www.automodel.com</recipient-description>
        </same>
      </RECIPIENT>
      <RETENTION>
       <EXTENSION  xmlns:xf="http://www.w3.org/TR/2002/WD-xquery-operators-20020816/#" >
          <xf:dt-yearMonthDuration>
           P2Y
          </xf:dt-yearMonthDuration>
       </EXTENSION>
        <stated-purpose/>
      </RETENTION>
    </STATEMENT>
    <STATEMENT>
      ...
    </STATEMENT>

  </POLICY>
```

4.2.3.2 An Example of the Customer Record Instance Segment

The following segment is of an instance of a customer record in XML format of Automart to which the resulting Organization Privacy Control Policies (OPCPs) in XACML obtained by the mapping can be applied:

```
<CustomerRecord id="dianshu" xmlns="urn:example:automart:schemas:record"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
 <!-- customer account infor -->
 <account>
  <id>dianshu</id>
  <password>123456</password>
  <email>hudianshu@hotmail.com</email>
  <timestamp>2011-02-01</timestamp>
 </account>
 ...
</record>
```

4.2.3.3 <p3p: STATEMENT>

As is demonstrated in the P3P policy example provided in 4.2.3.1, a P3P <STATEMENT> element specifies privacy promises in terms of the following: the collection of the specific kinds of UPI of a UPIG; the purposes for which the collected UPI of the UPIG can be accessed by the system entities or employees with particular roles within the e-business; the involved third-party organization recipients; and the time period for the retention of the UPI collection. The specific kinds of UPI to collect are described in the contained <DATA-GROUP> element. The associated UPIG is indicated in the contained <STATEMENT-GROUP> element. The bindings between the purposes and the roles are captured in the contained <PURPOSE-ROLES> element. The contained <RECIPIENT> element captures the identities of the organization recipients. The time period of the UPI collection retention is specified in the contained <RETENTION> element. In addition to the privacy promises specified in the <STATEMENT> element of the example, the privacy promise that describes customers' access to the UPI collection is specified in the <ACCESS> element in the containing <Policy> element.

Meanwhile, an XACML rule describes a privacy control practice in terms of the involved subjects, resources, environments, actions and the relations between them. The attributes of target subjects, resources, environments and actions are captured in the <Subject>, <Resource>, <Environment> and <Action> elements respectively. The relations between them are captured in the <Condition> element.

Hence it is intuitive that a P3P statement is equated to a XACML rule. The underlying idea of mapping a <p3p:STATEMENT> element to the corresponding XACML <Rule> element is explained hereunder. The <Resource> and <Subject> elements of the XACML rule can be formed based on the information extracted respectively from the <p3p:DATA-GROUP> and from the <p3p:RECIPIENT> and contained roles of the <p3p:PURPOSE-ROLES>. Since the employees of the organization recipients are not the data owners of the collected UPI to which they have access, "read" is the only action that they are allowed to perform on the data. In addition, the information used as the value of the action:purpose attribute is extracted from the <p3p:PURPOSE-ROLES> element. Finally, the timestamp included in <DATA-GROUP>, the time period included in <p3p:RETENTION> and the bindings between the roles and the purposes included in <p3p:PURPOSE-ROLES> are used to specify the <Condition> element of the XACML rule.

In general, a P3P <STATEMENT> is considered to be *relevant* to the mapping and is thus transformed to the corresponding XACML permit rule only if it satisfies all the following conditions: 1) it does not include <NON-IDENTIFIABLE/>; 2) its <DATA-GROUP> element refers to a UPIG or to some kinds of UPI of the UPIG; and 3) its <RETENTION> element contains any allowable element other than <no-retention/>.

P3P input pattern:

```
<STATEMENT>
        <EXTENSION>
                <p3p11:STATEMENT-GROUP id="account" />
        </EXTENSION>
      ...
</STATEMENT>
```

XACML output pattern:

```
<Rule RuleId="urn:oasis:names:tc:xacml:2.0:example:ruleid:account" Effect="Permit">
...
</Rule>
```

4.2.3.3.1 <p3p:DATA-GROUP>

In a relevant P3P <STATEMENT> element, the related UPIG is indicated by the ID attribute in the contained <STATEMENT-GROUP> element. The contained <DATA-GROUP> element refers either to all the related kinds of UPI of the UPIG or to some of them. If it refers to some of the related kinds of UPI of the UPIG, the UPIG name and the names of the UPI types involved are used together to generate the XPath expressions that locate the UPI. Note that the timestamp of the http post request itself is not identified data. It is stored in the user online account and is used for checking the expiry of the UPI retention only. Therefore it does not need to be specified in a <Resource> element. Its use is explained in detail in section 4.2.3.3.4.

P3P input pattern:

```
<STATEMENT>
        <EXTENSION>
                <p3p11:STATEMENT-GROUP id="account" />
        </EXTENSION>
        <CONSEQUENCE>We collect your email address to create your online account and process your order. You can
choose to receive flyers by email</CONSEQUENCE>
        <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
        <DATA ref="#user.home-info.online.email"/>
        <DATA ref="#dynamic.clickstream.timestamp"/>
      </DATA-GROUP>
      ...
</STATEMENT>
```

XACML ouput pattern:

```
   <Resource>
    <ResourceMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
      urn:example:Automart:schemas:record
     </AttributeValue>
    <ResourceAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:resource:target-namespace"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
    </ResourceMatch>
    <ResourceMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:xpath-node-match">
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
```

```
        /CustomerRecord/account/email
      </AttributeValue>
      <ResourceAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:resource:xpath"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
    </ResourceMatch>
  </Resource>
```

In the other case, if the <DATA-GROUP> element refers to all the related types of UPI of the UPIG, the UPIG name is used to generate the XPath expression which locates the UPI.

P3P input pattern:

```
<STATEMENT>
        <EXTENSION>
                <p3p11:STATEMENT-GROUP id="account" />
        </EXTENSION>
        <CONSEQUENCE>We collect your email address to create your online account and process your order. You can
choose to receive flyers by email</CONSEQUENCE>
        <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
        <DATA ref="#user.login.id"/>
        <DATA ref="#user.login.password"/>
        <DATA ref="#user.home-info.online.email"/>
        <DATA ref="#dynamic.clickstream.timestamp"/>
      </DATA-GROUP>
      ...
</STATEMENT>
```

XACML ouput pattern:

```
  <Resource>
    <ResourceMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
        urn:example:Automart:schemas:record
      </AttributeValue>
    <ResourceAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:resource:target-namespace"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
    </ResourceMatch>
    <ResourceMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:xpath-node-match">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
        /CustomerRecord/account
      </AttributeValue>
      <ResourceAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:resource:xpath"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
    </ResourceMatch>
  </Resource>
```

4.2.3.3.2 <p3p: PURPOSE-ROLES >

The <PURPOSE-ROLES> element contains purpose value elements which in turn contain the associated roles. The purpose elements are used as the values of the action:purpose attribute, while the associated roles are used to specify the system entities that and the group of employees who can access the resource. Note that all the target actions are "read" since the subjects (*i.e.* the employees of Automart) are not the data owners. The bindings between purposes and roles are specified in the <Condition> element. Note that, if there is only one purpose value element included in the <PURPOSE-ROLES> element and if the former contains only one role element,

the binding between them does not need to be specified under the <Condition> element because the logical relationship between <Subjects> and <Actions> in the <Target> element is **conjunctive.**

P3P input pattern:

```
<p3ppee:PURPOSE-ROLES>
 <p3ppee:marketing>
  <p3ppee:role>EmailFlyerServer</p3ppee:role>
 </p3ppee:marketing>
</p3ppee:PURPOSE-ROLES>
```

XACML output pattern:

```
<Target>
  <Subjects>
   <Subject>
    <SubjectMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
     EmailFlyerServer
     </AttributeValue>
     <SubjectAttributeDesignator  AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:role"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
    </SubjectMatch>
   </Subject>
  </Subjects>
<Actions>
   <Action>
    <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
     read
     </AttributeValue>
     <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
DataType="http://www.w3.org/2001/XMLSchema#string" />
    </ActionMatch>
    <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
     marketing
     </AttributeValue>
     <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:2.0:action:purpose"
DataType="http://www.w3.org/2001/XMLSchema#string" />
    </ActionMatch>
  </Action>
 </Actions>
</Target>
<Condition>
 <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:or">
  <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:and">
   <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
    <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">EmailFlyerServer</AttributeValue>
    <SubjectAttributeDesignator  AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:role"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
   </Apply>

   <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-equal">>
    <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">marketing</AttributeValue>
    <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:2.0:action:purpose"
DataType="http://www.w3.org/2001/XMLSchema#string" />
```

```
    </Apply>
   </Apply>
  </Apply>
</Condition>
```

4.2.3.3.3 <p3p: RECIPIENT>

The <RECIPIENT> element contains the identities of the organization recipients. They are used to specify the issuers of the role attributes of the authorizable employee groups. Note that the organization recipients who fall into the <same> category are the third-party recipients of the UPI collection. Their employees/systems can access the UPI only for the same marketing purpose as the first party but not for any other purpose of the first party.

P3P input pattern:

```
<RECIPIENT>
 <ours>
  <recipient-description>www.automart.com</recipient-description>
 </ours>
 <same required="opt-out">
  <recipient-description>www.automodel.com</recipient-description>
 </same>
</RECIPIENT>
```

XACML output pattern:

```
<Target>
  <Subjects>
   <Subject>
    <SubjectMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
     EmailFlyerServer
     </AttributeValue>
     <SubjectAttributeDesignator  AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:role"
DataType="http://www.w3.org/2001/XMLSchema#string" Issuer="www.automart.com"/>
    </SubjectMatch>
   </Subject>
   <Subject>
    <SubjectMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
     EmailFlyerServer
     </AttributeValue>
     <SubjectAttributeDesignator  AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:role"
DataType="http://www.w3.org/2001/XMLSchema#string" Issuer="www.automodel.com"/>
    </SubjectMatch>
   </Subject>
  </Subjects>
<Actions>
   <Action>
    <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
     read
     </AttributeValue>
     <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
DataType="http://www.w3.org/2001/XMLSchema#string" />
    </ActionMatch>
```

```
      <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
       <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
       marketing
       </AttributeValue>
       <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:2.0:action:purpose"
DataType="http://www.w3.org/2001/XMLSchema#string" />
      </ActionMatch>
     </Action>
  </Actions>
</Target>
<Condition>
  <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:or">
   <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:and">
    <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">EmailFlyerServer</AttributeValue>
     <SubjectAttributeDesignator    AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:role"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
    </Apply>

    <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-equal">>
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">marketing</AttributeValue>
     <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:2.0:action:purpose"
DataType="http://www.w3.org/2001/XMLSchema#string" />
    </Apply>
   </Apply>
  </Apply>
</Condition>
```

4.2.3.3.4 <p3p: RETENTION>

The <RETENTION> element specifies the time period of the UPI retention of the company. The time period is used in the condition element to check whether the retention has expired. If the retention has expired, access to the UPI is not permitted (not denied). Note that, if the <RETENTION> element includes <no-retention/> which means that the user private data to be collected will be discarded right after being used in a single online interaction and will not be stored on the website, the containing <STATEMENT> will not be transformed into the corresponding XACML rule. If the <RETENTION> element contains either <stated-purpose/>, <legal-requirement/> or <business-practices/>, the time period specified in the contained <EXTENSION> element is extracted and used in the <Condition> element to check whether the retention of the requested UPI has expired. To achieve this, the timestamp of the http post request that submits a user's UPI to create the online account for the user needs to be collected as shown in 4.2.3.1. As explained earlier in 4.2.2.3.3, if the <RETENTION> contains <indefinitely/>, the time period of retention is considered to be infinity. In this case, the <RETENTION> element is not used in the mapping since there is no need for checking the expiry of user private data retention.

P3P input pattern:

```
<p3p:RETENTION>
 <EXTENSION  xmlns:xf="http://www.w3.org/TR/2002/WD-xquery-operators-20020816/#" >
  <xf:dt-yearMonthDuration>
```

```
   P2Y
   </xf:dt-yearMonthDuration>
  </EXTENSION>
  <business-practitices/>
</p3p:RETENSION>
```

XACML output pattern:

```
<Condition>
    ,,,   FunctionId="urn:oasis:names:tc:xacml:1.0:function:and">
   <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:or">
    <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:and">
     <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">EmailFlyerServer</AttributeValue>
      <SubjectAttributeDesignator    AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:role"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
     </Apply>

     <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-equal">>
      <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">marketing</AttributeValue>
      <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:2.0:action:purpose"
DataType="http://www.w3.org/2001/XMLSchema#string" />
     </Apply>
    </Apply>
   </Apply>
   <!-- ********** The following part check the expiry of retention********** -->
    <Apply FunctionId="urn:oasis:names:tc:xacml:2.0:function:date-less-equal">
     <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:date-one-and-only">
      <EnvironmentAttributeDesignator AttributeId= "urn:oasis:names:tc:xacml:1.0:environment:current-date"
DataType="http://www.w3.org/2001/XMLSchema#date"/>
     </Apply>
     <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:date-add-yearMonthDuration">
      <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:date-one-and-only">
       <AttributeSelector RequestContextPath= "//CustomerRecord/account/timestamp/text()"
DataType="http://www.w3.org/2001/XMLSchema#date"/>
      </Apply>
      <AttributeValue DataType="http://www.w3.org/TR/2002/WD-xquery-operators-20020816#yearMonthDuration">
       <xf:dt-yearMonthDuration>
        P2Y
       </xf:dt-yearMonthDuration>
      </AttributeValue>
     </Apply>
   </Apply>

</Condition>
```

4.2.3.3.5 <p3p:NON-IDENTIFIABLE>

If a <NON-IDENTIFIABLE> element is included in the <STATEMENT> element, then the data elements in the <DATA-GROUP> element must refer either to non-identifiable data or to the identified/identifiable data that is anonymized upon collection (*e.g.* a web page collects only the city part of user postal address). In either case, the containing <STATEMENT> is not transformed into the corresponding XACML rule.

4.2.3.3.6 <p3p:CONSEQUENCE>

The <CONSEQUENCE> element provides a short explanation in human readable format of the privacy practices of the containing <STATEMENT> element to help users to understand the embedded privacy promises. Thus it is not used in the mapping.

4.2.3.4 <p3p:POLICY>

The <POLICY> element is the container of the <STATEMENT> elements specified therein. The corresponding XACML element is <Policy>. A P3P Policy element is considered relevant to the mapping and is thus transformed into the corresponding <Policy> only if it satisfies all the following conditions: 1) it contains at least one relevant <STATEMENT> element; 2) it does not contain the <TEST/> element; 3) its <ACCESS> element does not contain the <nonident/> tag; and 4) it is not expired. Note that the generated <Policy> element always has a RuleCombiningAlgId that is equal to "permit-overrides".

P3P input pattern:

<POLICY discuri=*"/privacy_policy.jsp"* name=***"RegisterPage"*** opturi=*"/privacy_policy.jsp"*>
<EXTENSION>
<p3p11:STAEMENT-GROUP-DEF id=*"account"* consent = *"mixed"* short-description=*"Account Registration Necessaries"*/>
...
</EXTENSION>
</POLICY>

XACML output pattern:

<Policy PolicyId="urn:oasis:names:tc:xacml:2.0:example:policyid:**RegisterPage**"
xmlns="urn:oasis:names:tc:xacml:2.0:policy:schema:os"
xmlns:xacml-context="urn:oasis:names:tc:xacml:2.0:context:schema:os"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:oasis:names:tc:xacml:2.0:policy:schema:os http://docs.oasis-open.org/xacml/access_control-xacml-2.0-policy-schema-os.xsd"
xmlns:xf="http://www.w3.org/TR/2002/WD-xquery-operators-20020816/#"
xmlns:am="http:www.automart.com/schemas/record.xsd"
RuleCombiningAlgId="urn:oasis:names:tc:xacml:1.0:rule-combining-algorithm:**permit-overrides**">
...
</Policy>

4.2.3.4.1 <p3p:TEST/>

The presence of the <TEST/> element indicates that the containing policy is just an example and must not be considered to be a valid P3P policy. If a <POLICY> element contains this element, it is considered irrelevant to the mapping regardless of whether its <STATEMENT> elements are relevant.

4.2.3.4.2 <p3p:ACCESS>

As explained in the beginning of section 4.2.3, the definition of the <ACCESS> element is ambiguous and therefore only the element that contains <nonident/>, <all/> or <none/> is relevant to the mapping. If the <ACCESS> element contains <nonident/> which means that the website does not collect identified data, the containing <POLICY> is considered irrelevant to the mapping. If it contains <all/>, an additional XACML rule such as the one shown in the XACML output pattern set out below needs to be formed in order to enable the customers to access their private data. Note that a customer is allowed to perform "read" and "write" actions on his/her own private data because he/she is the owner of the data. The case of <none/> is automatically handled by the XACML rules generated by the mapping because there is no rule in the resulting XACML policy that permits access by customers to their private data and, thus, access is indirectly denied.

P3P input pattern:

```
<ACCESS>
 <all/>
</ACCESS>
```

XACML output pattern:

```
<Rule RuleId="urn:oasis:names:tc:xacml:2.0:example:ruleid:customer" Effect="Permit">
 <Target>
  <Resources>
   <Resource>
    <ResourceMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
      urn:example:Automart:schemas:record
     </AttributeValue>
     <ResourceAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:resource:target-namespace"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
    </ResourceMatch>
    <ResourceMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:xpath-node-match">
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
      /CustomerRecord/
     </AttributeValue>
     <ResourceAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:resource:xpath"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
    </ResourceMatch>
   </Resource>
  </Resources>
  <Actions>
   <Action>
    <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
     read
     </AttributeValue>
     <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
DataType="http://www.w3.org/2001/XMLSchema#string" />
    </ActionMatch>
   </Action>
   <Action>
    <ActionMatch MatchId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
     <AttributeValue DataType="http://www.w3.org/2001/XMLSchema#string">
     write
```

```
      </AttributeValue>
      <ActionAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:1.0:action:action-id"
DataType="http://www.w3.org/2001/XMLSchema#string" />
    </ActionMatch>
   </Action>
  </Actions>
 <Target/>
 <Condition>
  <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-equal">
   <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-one-and-only">
    <SubjectAttributeDesignator AttributeId="urn:oasis:names:tc:xacml:2.0:example:attribute:customer-id"
DataType="http://www.w3.org/2001/XMLSchema#string"/>
   </Apply>
   <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:function:string-one-and-only">
    <AttributeSelector RequestContextPath="//xacml-context:Resource/xacml-
context:ResourceContent/CustomerRecord/account/id/text()" DataType="http://www.w3.org/2001/XMLSchema#string"/>
   </Apply>
  </Apply>
 </Condition>
</Rule>
```

4.2.3.4.3 Unusable P3P Elements of <POLICY>

<p3p:ENTITY>

The <ENTITY> element provides a precise description of the legal entity making the representation of the privacy promises contained in the <POLICY> element. The name of the organization contained in this element could potentially be used as the issuer of the subject, resource and action attributes in the resulting XACML policy. However, for the purpose of this mapping, the website URLs specified in the <RECIPIENT> element are instead used to refer to the identity of the company and to the identities of the third-party organization recipients.

<p3p:DISPUTES>

The <DISPUTES> element provides dispute-resolution procedures that may be followed in case of a privacy dispute. This element is irrelevant to the mapping since the information included therein does not add any value in refining the Organizational Privacy Control Policies (OPCPs) generated by using the mapping.

4.2.3.5 Generating <PolicySet>

After applying the mapping, the Organizational Privacy Control Policies (OPCPs) are derived based on the privacy policy written in P3P with P3PPEE. Then a <PolicySet> which aggregates the resulting OPCPs can be easily formed as is shown in the XACML pattern below.

XACML output pattern:

```
<PolicySet xmlns="urn:oasis:names:tc:xacml:2.0:policy:schema:os" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="urn:oasis:names:tc:xacml:2.0:policy:schema:os http://docs.oasis-open.org/xacml/access_control-
```

```
xacml-2.0-policy-schema-os.xsd" PolicySetId= "urn:oasis:names:tc:xacml:2.0:example:policysetid:OPCP"
PolicyCombiningAlgId="urn:oasis:names:tc:xacml:1.0:policy-combining-algorithm:permit-overrides">
 <PolicyIdReference>
  urn:oasis:names:tc:xacml:2.0:example:policyid: RegisterPage
 </PolicyIdReference>
 ...
</PolicySet>
```

4.2.4 Privacy Practice Negotiation Mechanism

Based on the accurate mapping from P3P to XACML explained in section 4.2.3, a privacy practice negotiation mechanism between the EAPEX enabled company and the web users is developed and explained in this section. The following scenario describes how it works.

Firstly an e-commerce website publishes its privacy policies in P3P with P3PPEE. Then a user interacts with his EAPEX enabled user agent to store his/her identified data, to set User Privacy Preferences (UPPs) and to generate the corresponding APPEL preference file. After that the user visits the website. The user agent then automatically retrieves the privacy policies and prompts the user the privacy promises (*i.e.* the <STATMENT> elements) contained in the policy that covers the page currently being visited. When the user visits a web page that requires him/her to submit the UPIGs of his/her identified data, the user agent automatically finds the <STATEMENT> elements that contradict the UPPs. The user agent then negotiates the privacy practices with the website by extracting the contradicting parts (*i.e.* <PURPOSE-ROLES> and <RECIPIENT>) and the names of the relevant UPIGs and by generating the XACML rules (*i.e.* UPCP rules) that *deny* the contradicting privacy practices. Then the user agent fills in the required UPIGs and submits them together with the resulting UPCP rules to the website. When the web server receives the http post request, it extracts the UPIGs and the UPCP rules and stores them together in the UPI and UPP storage.

In the company, when a request for access to a UPIG of a particular user is sent to the Policy Decision Point (PDP), the Policy Information Point (PIP) dynamically generates a UPCP that includes the related UPCP rule of the user and sends the UPCP to the PDP. The PDP then dynamically generates a <PolicySet> which includes both the <PolicySet> of the OPCPs (section 4.2.3.5) and the UPCP, and evaluates the resulting <PolicySet> (*i.e.* the PCP) for authorizing decision-making. Note that the resulting <PolicySet> element must use "deny-overrides" as the policy combining algorithm to ensure that the deny effect of the UPCP overrides the permit effect of the OPCPs in order to achieve the enforcement of the negotiated privacy practices.

The mapping used by the EAPEX user agent to generate UPCP rules is exactly the same as the one used to generate OPCPs except for the following two differences: 1) the resulting UPCP rules must have "deny" instead of "permit" as the rule effect; and 2) the contradicting <RETENTION> value element is not used in the mapping. The contradicting <RETENTION> element is considered irrelevant to the mapping used by the negotiation mechanism because the

user should trust an EAPEX e-business as long as it clearly states the time period of UPI retention. If the website uses <indefinitely/>, the user agent can inform the user that his/her UPI may be retained permanently. Thus the user can simply turn to another EAPEX e-business that provides similar services.

4.2.5 Audit

In order to verify a website's compliance with EAPEX – that is, to verify that the Organizational Privacy Control Policies (OPCPs) used by the Privacy Enforcement Model (PEM) are the same as those derived by transforming the Web Privacy Policies (WPPs) and to verify that the User Privacy Control Policy (UPCP) rules used by the PEM are the same as those submitted by the users – a trusted third party (TTP) serving as the external auditor is included in EAPEX. Given the fact that the TTP is responsible for checking the consistency between the WPPs and the OPCPs of an EAPEX enabled website and the consistency between the User Privacy Control Policy (UPCP) rules submitted by the users and those stored in the UPI and UPP storage of the website, it follows that the verification mechanism must consist of two procedures. These procedures are explained hereunder.

4.2.5.1 Checking Consistency between OPCPs and WPPs

The TTP makes use of the same XSLT stylesheet as the one used by the company and an XSLT engine to transform the WPPs into the corresponding OPCPs. The consistency can then be checked by comparing the hashes of the generated OPCPs and those of the OPCPs used by the company.

4.2.5.2 Checking Consistency between the UPCP Rules Generated by the User Agents and Those Stored on the Website

Whenever the EAPEX user agent submits the UPIGs of the user's UPI along with the generated UPCP rules to an EAPEX enabled website, the user agent also sends a tuple of the user ID (*i.e.* the username used in the online account of the website), the identity of the website (*i.e.* the website URL), the UPCP rule ID and the hash of the UPCP rule to the TTP for each of the UPCP rules. When the TTP checks the consistency between the UPCP rules stored on the website and those submitted by the user agents, the TTP compares the hashes of the UPCP rules stored on the website with those stored in the TTP for each user. If the hashes match for all the users, the TTP can confirm that the UPCP rules are consistent with those submitted by the user agents.

4.2.5.3 Necessity of On-site Verification

Since the size of the UPCP rules that need to be checked can be extremely large, the verification mechanism of the TTP is not always feasible to be invoked through online services. Moreover, the verification by the TTP through online services is not sufficient as a website can send the consistent OPCPs and the UPCP rules to the TTP while in practice using modified OPCPs and UPCP rules which are not even used in the worst case scenario. Therefore, the TTP needs to perform on site investigation to verify a website's compliance with EAPEX on a periodic basis.

4.2.5.4 Certifying the Compliance with EAPEX

After the TTP has verified a website's compliance with EAPEX by using the verification mechanism explained above, it can make use of any public key encryption system (*e.g.* RSA) to issue a certificate for the website that is valid until the time of the next verification. This ensures that, when a user visits the website, he/she can easily check the validity of the certificate by using the public key of the TTP.

4.2.6 Conclusion of EAPEX

In an e-business environment, privacy enforcement can be achieved only when the privacy promises of the e-business and the privacy preferences of the users are used together in the privacy control mechanism. In section 4.2, the Enhanced Architecture for Privacy Enforcement using XML (EAPEX) provides an architecture design to achieve privacy enforcement. This architecture design addresses users' privacy concerns and also enforces e-businesses' privacy promises. In addition, with all the complementary mechanisms proposed in section 4.2, EAPEX can be implemented using existing technologies. Chapter 5 documents the EAPEX website "Automart" and the EAPEX user agent "Privacy Controller" implemented in my thesis research.

4.3 Architecture for Privacy Enforcement using Policy Based Encryption

This section explains the theoretical design of a new architecture for privacy enforcement that uses the Policy Based Encryption (PBE) system introduced by Bagga and Molva, P3P with P3PPEE, and APPEL together. The new architecture is called the Architecture for Privacy Enforcement using PBE (APEP). APEP can be used as an alternative to enforce privacy in an e-business environment.

4.3.1 The Policy Based Encryption (PBE) System

The Policy Based Encryption (PBE) system proposed by Molva and Bagga based on bilinear pairings over elliptic curves looks very promising for enforcing privacy in a distributed e-business environment where multiple trusted authorities are allowed to participate in the authorization process. As stated in [39], the PBE system can be used for access control and therefore it can be used as an alternative to a traditional access control mechanism like XACML to enforce privacy. A major advantage of this PBE system is that it allows credentials to be kept secret by their owner while proving policy compliance in contrast to the traditional approach in which credentials have to be revealed.

Garson and Adams proposed the theoretical architecture design of a privacy enforcement system in an e-hospital environment. In their design, the extended PBE system which was originally proposed in [39] is used to enforce privacy. According to the researchers, the PBE system works naturally in the e-hospital setting. However, an e-business environment is very different from the e-hospital setting. In the health-care setting, a patient has few choices of hospital due to the location and time constraints. Although hospitals are responsible for formulating and enforcing proper privacy policies in order to achieve compliance with privacy statutes like PIPEDA and PHIPA, hospitals do not need to take patients' privacy preferences into consideration to prevent the loss of potential patients. Therefore, the PBE system itself is sufficient to enforce privacy in such a setting. The situation in an e-business environment is quite different. Nowadays web users have many choices of online service providers. Moreover, many e-business users are very cautious about giving out their private information because most of them have experience in receiving spam emails, physical junk mails and unsolicited phone calls from marketing companies. Therefore, e-businesses must take customer's privacy preferences seriously in order to prevent the loss of existing and potential customers. In such an environment, the PBE system must cooperate with other privacy-related technologies in order enforce privacy.

The following section explains the theoretical design of the privacy-enforcement architecture proposed through my thesis research that integrates the PBE system, P3P with P3PPEE, and APPEL to enforce privacy in an e-business environment.

4.3.2 Architecture Design

Based on the policy-based encryption system explained in the previous section in combination with P3P, P3PPEE and APPEL, a new privacy enforcement architecture for a e-commerce settings is proposed in this section. It is shown in the following figure:

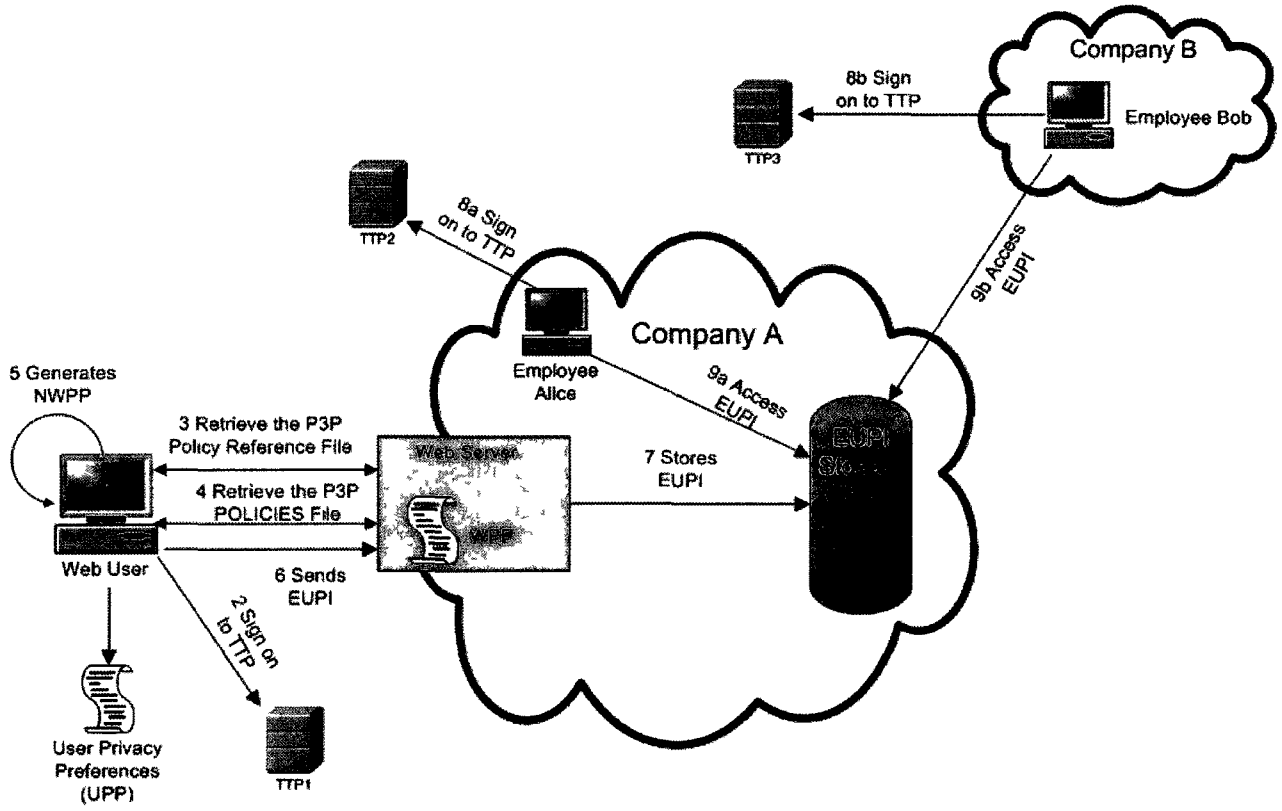


Figure 10 - Architecture Design Overview

The data flow shown in the above figure is explained as the follows: 1) the web user interacts with his/her user agent to set privacy preferences and generate a UPP file in APPEL; 2) the user signs on to the TTP on which he/she relies and obtains the credential of his ID for using on the website of company A; 3) upon visiting the website, the user agent sends an http request to retrieve the P3P policy reference file; 4) the user agent parses the P3P policy reference file and sends an http request to retrieve the POLICIES file that contains all the WPP policies covering the URL-space of the website; 5) once the user visits a web page that collects UPIGs, the user agent parses the WPP covering the page and removes the P3P elements (*e.g.* a particular purpose value tag like <marketing/> and the contained role tags) that contradict the UPP to derive the Negotiated Web Privacy Policy (NWPP); 6) for each UPIG required, the user agent transforms the related <STATEMENT> elements of the NWPP into the corresponding monotonic logical expressions and uses them to encrypt and send the user's UPI of the UPIG; 7) the web server stores the tuples of the user's ID in plaintext, the public key of the TTP on which the user relies or the URL for retrieve the public key, the encrypted UPIG and the associated logical expression in the Encrypted User Private Information (EUPI) storage; 8) Alice (8a) who is an employee of the company and Bob (8b) who is an employee of a third party sign on to TTP2 and TTP3 respectively; and 9) Alice (9a) and Bob (9b) access the EUPI and can retrieve the UPI only if their credential sets satisfy the policy. It should be noted that the EUPI storage stores the web

user's ID in plaintext format for indexing customer records. It should also be noted that the EUPI storage stores the public key of the TTP on which the user relies in order to ensure that only the user can make changes to the EUPI.

In order to enable the user agent to encrypt UPI, the public key of the TTP must be included in the WPP for each intended organization recipient. A possible way to do so is to use the <recipient-description> elements of the WPP.

4.3.3 Audit Logs

Audit logs allow the e-business to track its employees' activities on the EUPI storage system. If by any chance an employee accesses the EUPI storage in an unintended way, then that activity must be recorded and be made available for further analysis. For example, if a customer service representative is accessing a large amount of UPI in an unreasonably short time period, such activities must be recorded for investigation. It could be the case that a business spy who has infiltrated the workforce is stealing the customer information. In order to at least keep records of abnormal activities on the EUPI storage system, logging functionality must be added to APEP.

4.4 The Analysis of the Two Privacy Enforcement Architectures

This section provides the analysis of the two privacy enforcement architectures (EAPEX and APEP). The following table summarizes the characteristics of EAPEX and APEP:

	EAPEX	APEP
Web Privacy Policy	P3P with P3PPEE	P3P with P3PPEE
User Privacy Preference	APPEL	APPEL
Privacy Control Mechansim	XACML	Policy Based Encryption
Network Encryption	SSL	Policy Based Encryption
UPI Stored by E-business	plaintext	ciphertext
Depending on TTP	yes	yes
Requires Auditing	yes	no

Table 2 – Summary of EAPEX and APEP

In general, privacy enforcement in an e-business environment can be achieved by deploying either EAPEX or APEP. A major difference in terms of privacy control between EAPEX and APEP is that the privacy control mechanism used in EAPEX is XACML while the one used in

APEP is the PBE system. Therefore, in EAPEX, privacy practices are enforced on the e-business side by the PCM. Thus, external auditing must be performed on a periodic basis to ensure the e-business's compliance with EAPEX. On the other hand, in APEP, privacy practices are enforced by the user agent on the client side. Hence the external auditing is not necessary.

A major difference in terms of security between EAPEX and APEP is explained below. In APEP, the PBE system provides encryption in addition to privacy enforcement. Thus, the security of the web-based transactions involving UPI is automatically achieved. Moreover, because the UPI stored on the e-business side is in ciphertext format, APEP is invulnerable to attackers who manage to gain physical access to the database. On the other hand, in EAPEX, XACML only provides privacy enforcement but not encryption. Thus, network encryption mechanisms like SSL must be in place to ensure the security of the web-based transactions involving UPI. Note that the UPI received by the e-business is directly stored in plaintext format in the database and, as a result, EAPEX is vulnerable to physical attacks to the database.

APEP seems to be superior to EAPEX since the PBE system provides both encryption and privacy control in one package. However, APEP has a major drawback for handling key compromise. In case of a key compromise, updating WPP's EXPIRY element can be used to disable a requester's credentials for accessing the UPI encrypted under the new WPP which have been acquired and saved from previous decryption processes. All the UPI encrypted under the old WPP will also have to be updated. This can be done by the customers decrypting and then encrypting their UPI under the new policy. The updating operation can be very expensive and time-consuming and the associated overhead costs are exacerbated as the number of customers increases. In addition, WPP in APEP must be specified to be valid for a short time period in order to prevent a key compromise. The frequent updating operation makes the overhead problem more severe.

In general, both EAPEX and APEP achieve privacy enforcement in an e-business environment. The PBE system used in the APEP seems to be less complex than the combination of XACML and SSL. Thus APEP may be easier to implement and, in consequence, it may be more suitable than EAPEX in a small e-business setting. On the other hand, APEP does not fit into a large e-business setting because of the overhead costs which are incurred by the frequent updating operation and because of the amplification of these costs that results as the number of customers in the setting increases. Thus, EAPEX seems to be more appropriate in a large e-business setting.

Chapter 5 Implementation Details

This chapter documents the implementation of EAPEX – performed through this thesis research – which consists of the EAPEX website "Automart" and the EAPEX user agent "Privacy Controller". The bandwidth overhead incurred by deploying EAPEX in the implementation is illustrated at the end of this chapter.

5.1 Automart

5.1.1 Overview

Automart is a fully functional e-commerce website implemented on JEE that sells used vehicles. It is an EAPEX enabled website as the privacy policies that explain its privacy practices are specified in P3P 1.1 with P3PPEE. The privacy policies can be automatically and accurately transformed into the corresponding OPCPs by using the mapping introduced in section 4.2.3 in order to enforce the privacy control practices. In addition, when Automart collects the UPIGs through its registration page, it allows the user agent to submit the required UPIGs as well as the UPCP rules which specify the non-preferred privacy control practices regarding the OPCPs. The submitted UPCP rules can then used together with the OPCPs by the PEM (this not being implemented in this thesis research) to enforce the negotiated privacy control practices. The rest of section 5.1 is organized into four sections. Section 5.1.2 explains the technologies on which Automart is implemented, as well as the JEE design patterns which are implemented and deployed in Automart. Section 5.1.3 shows the screen shots of the application runtime. It also explains the data collection practices of the major pages of Automart. Section 5.1.4 explains the user private information collection practices and the EAPEX Compliance of Automart. Finally the privacy policies written in natural language are shown in Section 5.1.5.

5.1.2 Technologies Employed and Design Patterns

5.1.2.1 Technologies Employed

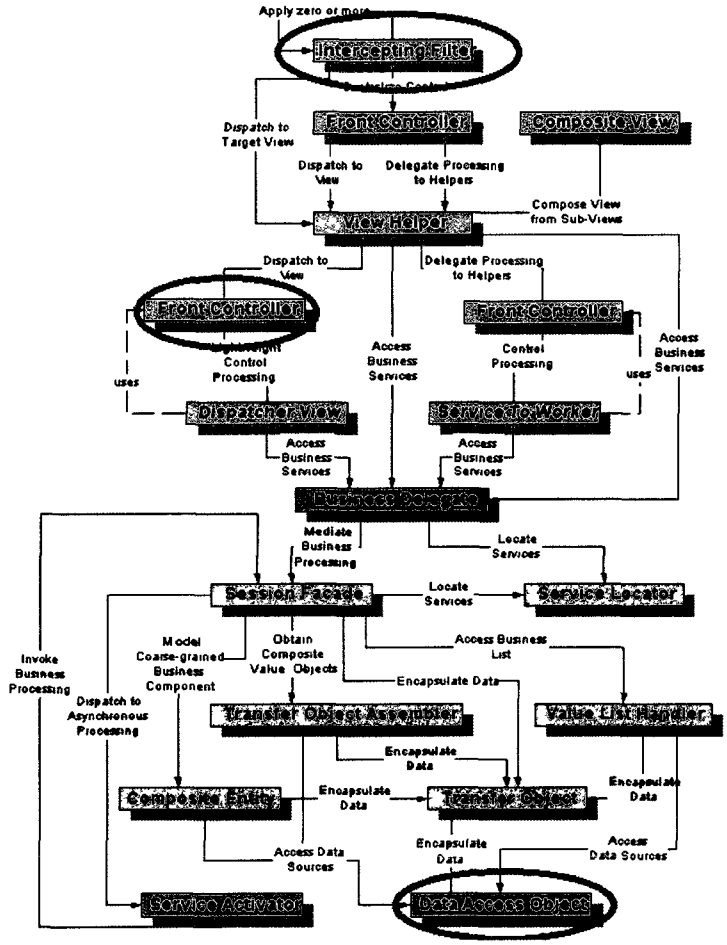
Web container: Tomcat 6.0, JDBC

Database: MySQL 5.1

IDE: NetBean 6.9.1

5.1.2.2 Design Patterns

The following diagram illustrates all JEE core design patterns (http://java.sun.com/blueprints/corej2eepatterns/Patterns/)



Several JEE core design patterns are deployed and implemented in Automart. As circled on the diagram above, the models implemented in Automart as listed as follows:

- **Generic Data Access Object Pattern**
  This is implemented and packed in the Automart web module. The DAO component is responsible for all communications with the database.

- **Facade Pattern**
  The facade pattern is implemented (ModelFacade.java) to provide a bulk behaviour repository. It handles calls from the business layer and retrieves inquired information using DAO.

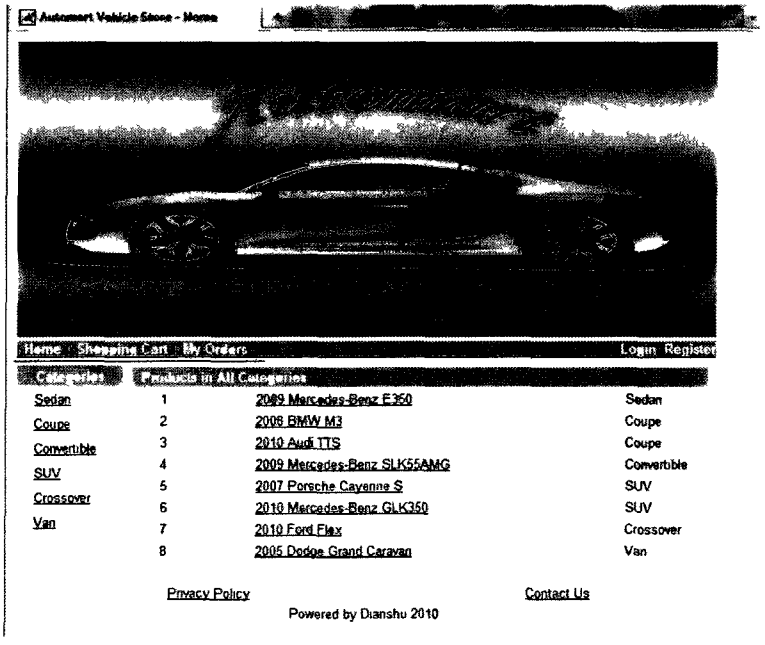
- **Intercepting Filter**
  Servlet Filter is a typical intercepting filter in JEE specification. In Automart, the Servlet Filter is implemented as a security measure to check whether the caller of a URL resource has correct permission (authentication).

- **Front Controller**
  JEE Servlet is used as a means of handling requests from the presentation and of delegating the request to the business layer. In Automart, a single servlet is implemented and deployed to handle various requests.

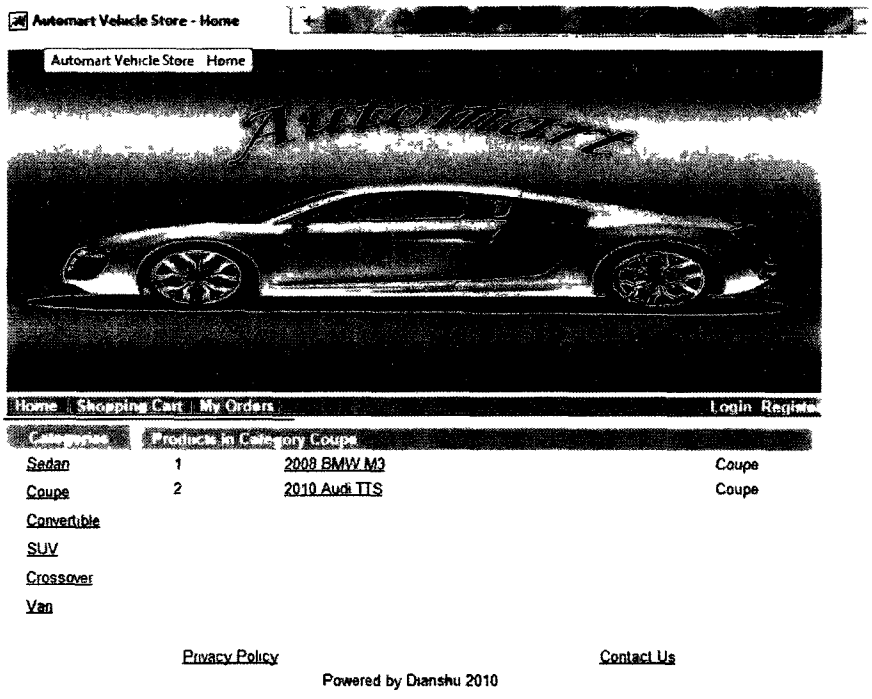
5.1.3 Data Collection Practices and Screen Shots of the Major Web Pages of Automart

5.1.3.1 index.jsp

This is the homepage of Automart. It lists all the vehicles in stock as shown by the screen shot below. This page collects clients' IP address and the timestamps of http requests for <p3p:develop/> and <p3p:admin/> purposes (*e g* to detect denial-of-service attacks).

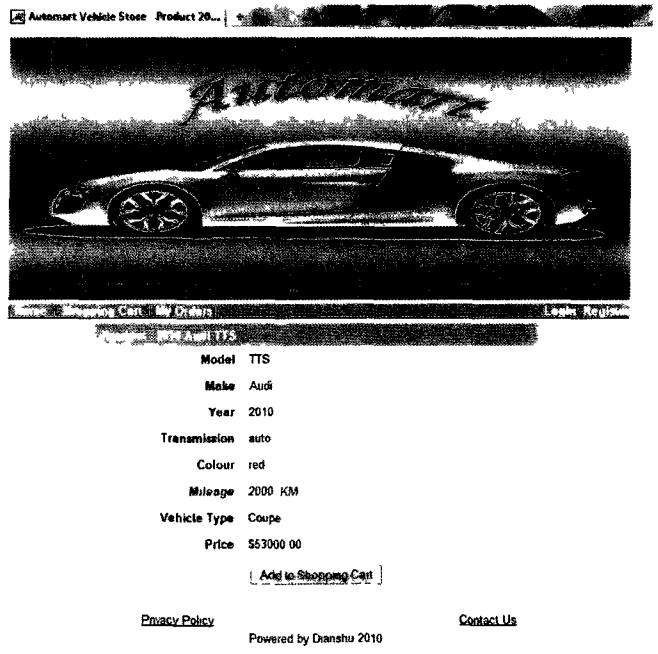


Users can browse the stock vehicles by category on this page as is shown in the screen shot below. Thus this page also collects the click stream to retrieve the vehicles of the category selected by user. Note that the click stream is used for a single online interaction (*i e* to retrieve vehicles by category) and is not stored thereafter.



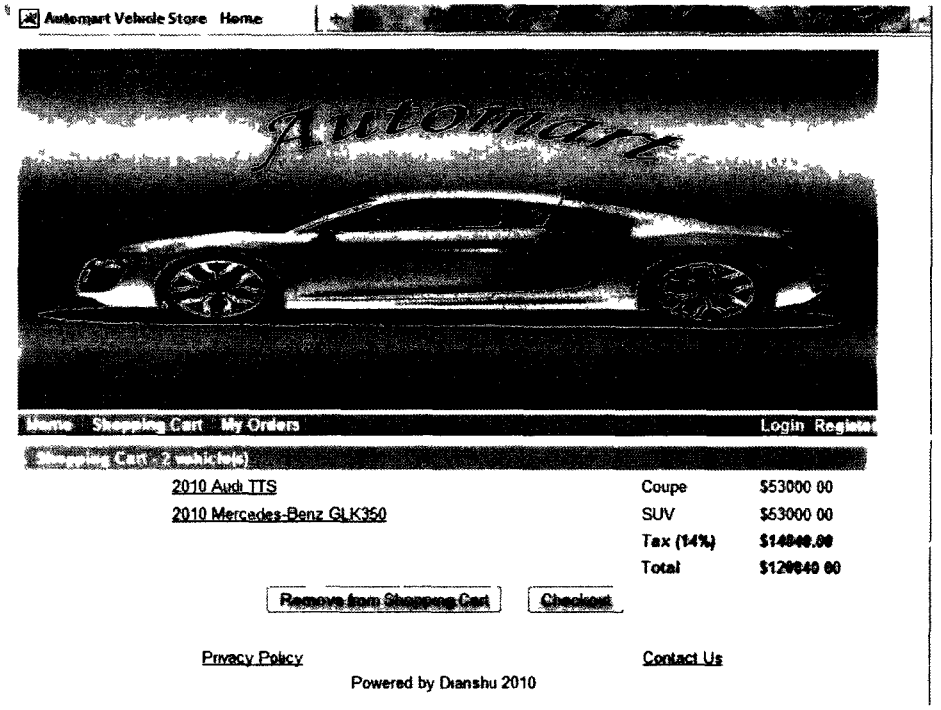
5.1.3.2 product.jsp

This page collects the query string part of the http request to retrieve the details of the vehicle chosen by user. It also uses the query string to recommend another vehicle that is popular and of the same category. The query string is discarded and not stored thereafter.

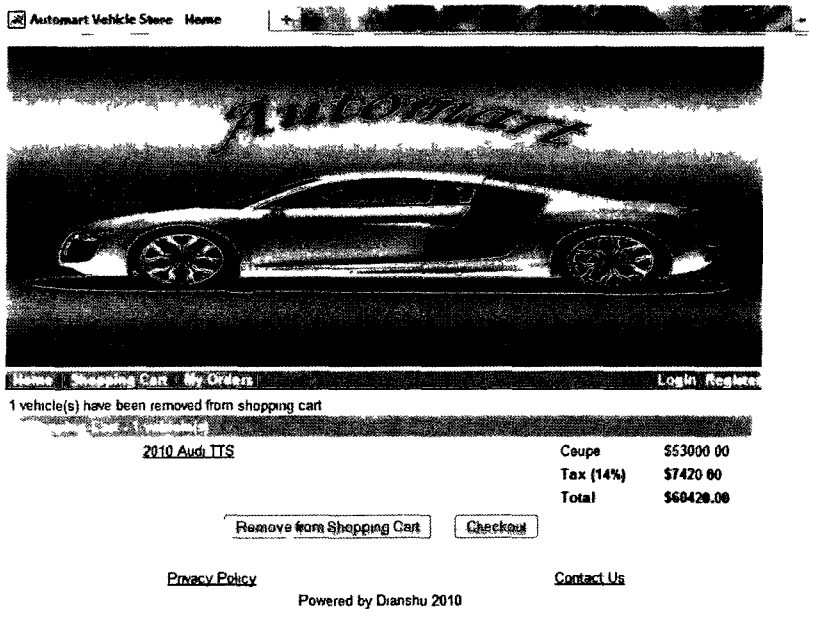


5.1.3.3 shopping_cart.jsp

This page shows the products in shopping cart.

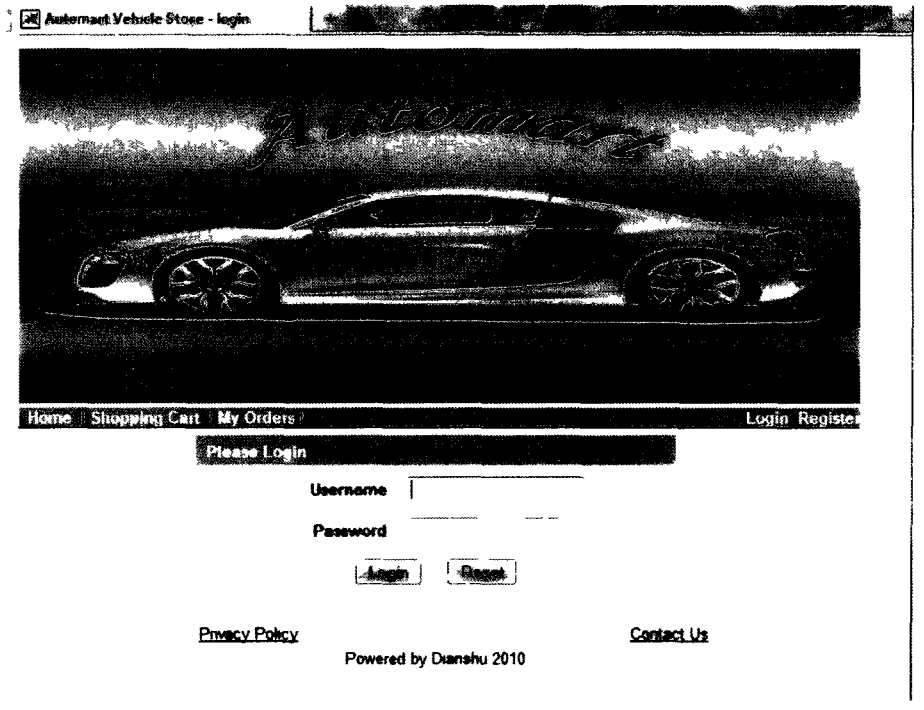


It also collects the IDs of the products (<p3p:other.httpmethod/>) chosen by customer to remove them from the shopping cart.

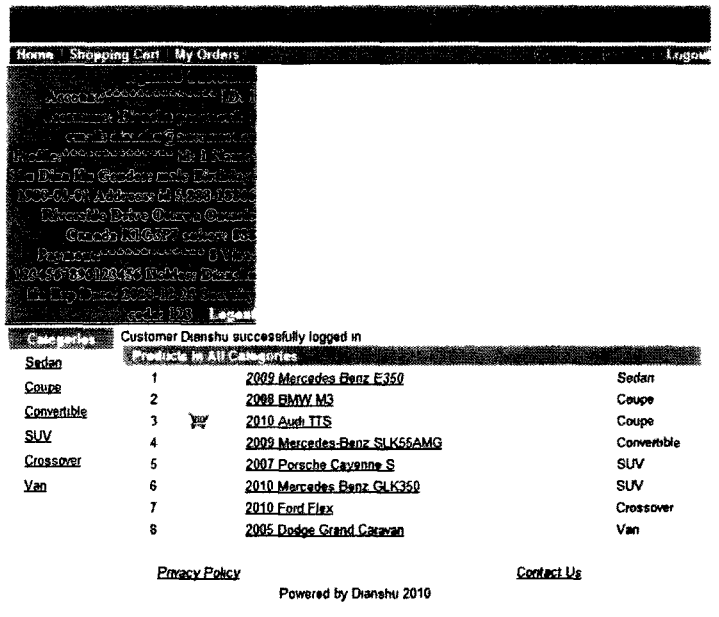


5.1.3.4 login.jsp

As shown in the screen shot below, this page collects the username and password (<p3p:login>) in order to authenticate registered customers. Note that the collected username and password are not stored after authentication.

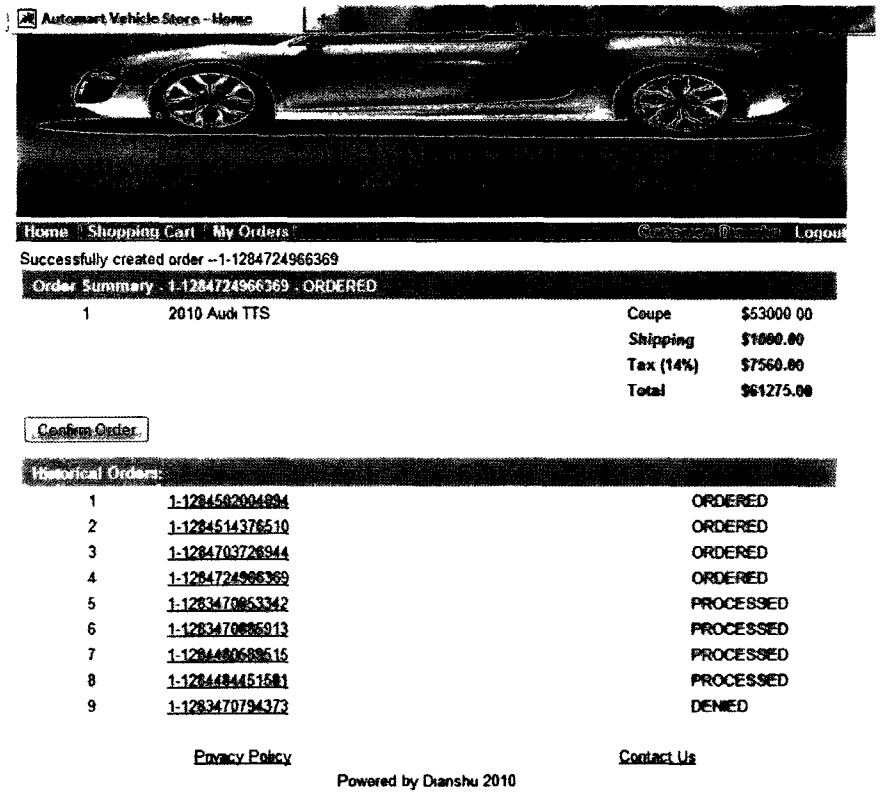


The screen shot below shows the homepage to which customers are redirected after successful authentication.

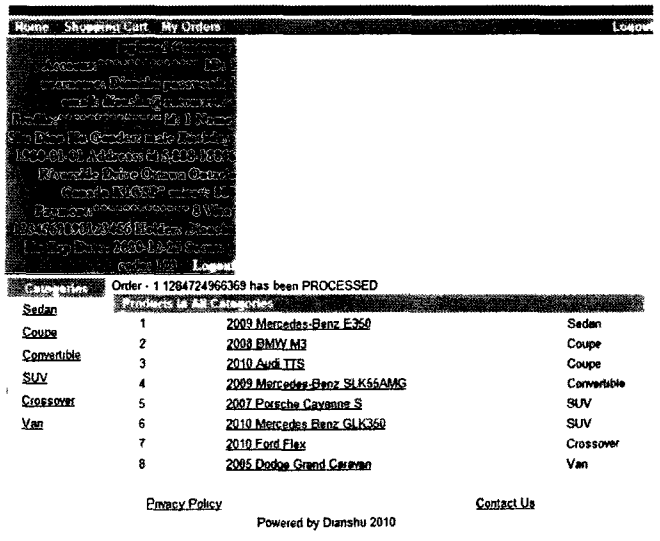


5.1.3.5 order.jsp

The order page shows a summary of the current order as shown in the screen shot below. This page also shows the previous orders that the user can review.

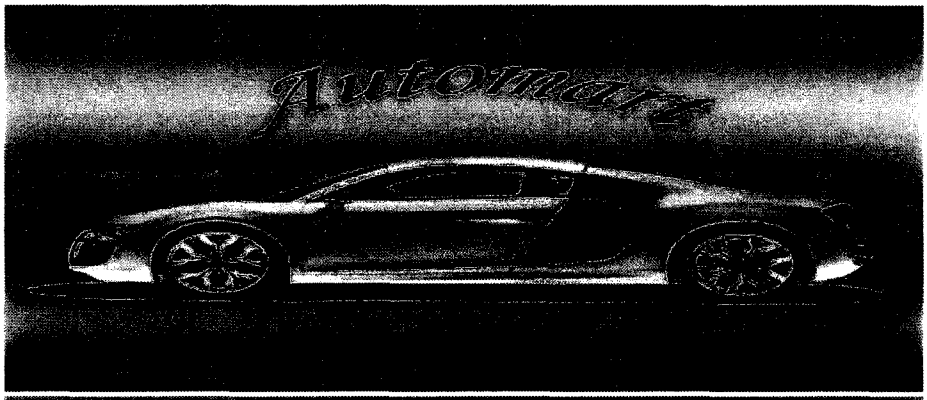


The user can submit this order by clicking the "confirm" button. Thus the time stamp (<p3p:timestamp>) is collected and used to generate the order ID. The screen shot below shows the home page to which customers are redirected after successful order submission.



5.1.3.6 historical_order_details.jsp

This page collects the query string part of the http request to retrieve the details of the historical order chosen by the user. The screen shot below shows the details of the historical order whose order ID is 1-1283470853342.



Home | Shopping Cart | My Orders | [illegible] Logout

Order Summary - 1-1283470853342 - PROCESSED

Order submission time: Thu Sep 02 19:40:53 EDT 2010

1	2010 Audi TTS	Coupe	$53000.00
		Shipping	**$1000.00**
		Tax (14%)	**$7280.00**
		Total	**$61275.00**

Billing and Shipping Address Information

Suite number	888
Street number	8866
Street name	Riverside Drive
City	Ottawa
Province	Ontario
Country	Canada
Postal code	K1G3P7

Payment Information

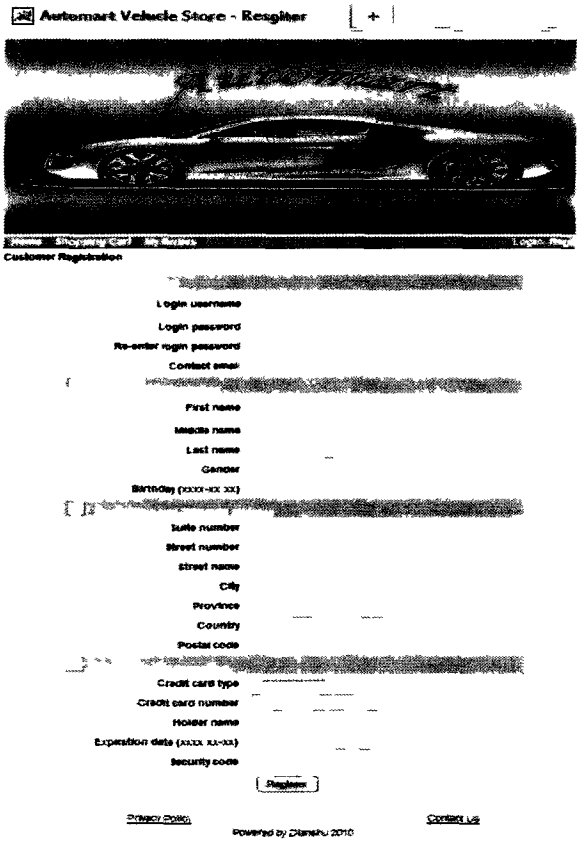
Credit card type	Visa
Credit card number	**********3456
Holder name	Dianshu Hu
Expiration date (xxxx-xx-xx)	2020-12-25
Security code	123

Privacy Policy Contact Us

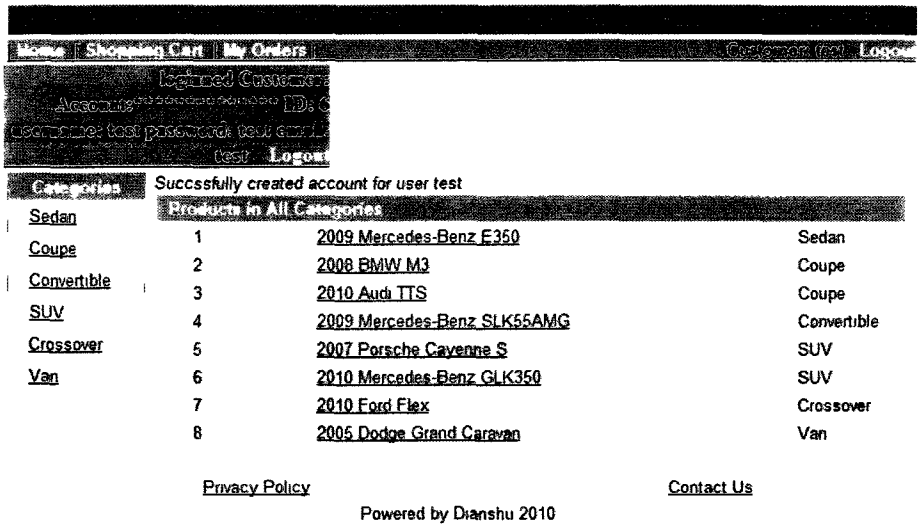
Powered by Dianshu 2010

5.1.3.7 register.jsp

This page requires users to submit various pieces of UPI whose types are denoted by the UPIGs in order to create an online account. There are four UPIGs defined in the database of Automart and exposed through the WPP. The UPIGs are also indicated by the registration page itself as is shown in the screen shot below. They are "account", "profile", "address" and "payment". The "account" UPIG consists of the following UPI types: <p3p:id/>, <p3p:password/>, <p3p:email/> and <p3p:timestamp/> . The "profile" UPIG consists of the following UPI types: <p3p:given/>, <p3p:middle/>, <p3p:family/>, <p3p:bdate/> and <p3p:gender/>. The "address" UPIG consists of the following UPI types: <p3p:street/>, <p3p:city/>, <p3p:stateprov/>, <p3p:postalcode/> and <p3p:country/>. Finally, the "payment" UPIG consists of the following UPI types: <p3ppee:cctype/> (*i.e.* credit card type), <p3ppee:ccnum/> (*i.e.* credit card number), <p3ppee:holdername/> (*i.e.* credit card holder name), <p3ppee:expiry/> and <p3ppee:securitycode/>.



The screen shot below shows the homepage to which customers are redirected after successful registration.



5.1.4 Private Information Collection Practices and EAPEX Compliance of Automart

The User Private Information (UPI) pieces of the data types included in the four UPIGs are collected on the registration page as explained in 5.1.3.7. The four UPIGs are "account", "profile", "address" and "payment". The <ACCESS> element of the P3P policy covering this page includes the <all/> tag as the registered users can login and update the UPI stored in their online accounts. The privacy practices of the UPI represented by the four UPIGs are explained in the subsections below.

5.1.4.1 The Privacy Practices of the User Private Information Group "account"

The privacy promises of non-marketing usage of the first party (Automart) regarding the UPI represented by the account UPIG are comprised of two parts. Firstly, the UPI represented by the account UPIG is stored for two years and is used by the web server of Automart to create online accounts for users. This UPI is also used by the web server and by the customer service representatives of Automart for communication purposes. Secondly, the UPI represented by the account UPIG as a whole is used by Automart only and is not shared with any other organization. The P3P <STATEMENT> element describing the privacy promises is shown as follows:

```
<STATEMENT>
        <EXTENSION>
                <p3p11 STATEMENT-GROUP id="account" />
        </EXTENSION>
        <CONSEQUENCE>We collect essential information to create an account for you and process your order  You need to provide a
username a password and your email address </CONSEQUENCE>
      <DATA-GROUP xmlns="http //www w3 org/2002/01/P3Pv1">
        <DATA ref="#user login id"/>
        <DATA ref="#user login password"/>
        <DATA ref="#user home-info online email"/>
```

```
    <DATA ref="#dynamic clickstream timestamp"/>
  </DATA-GROUP>

  <PURPOSE>
    <EXTENSION>
      <PPURPOSE xmlns="http //www w3 org/2004/02/P3Pv11">
        <account/>
      </PPURPOSE>
    </EXTENSION>
    <current/>
    <EXTENSION>
      <p3ppee PURPOSE-ROLES>
            <p3ppee account>
                  <p3ppee role>WebServer</p3ppcee role>
            </p3ppee account>
            <p3ppee communicate>
                  <p3ppee role>WebServer</p3ppcee role>
                  <p3ppee role>CSR</p3ppcee role>
            </p3ppee communicate>
      </p3ppee PURPOSE-ROLES>
    </EXTENSION>
  </PURPOSE>
  <RECIPIENT>
    <ours>
      <recipient-description>www automart com</recipient-description>
    </ours>
  </RECIPIENT>
  <RETENTION>
      <EXTENSION xmlns xf="http //www w3 org/TR/2002/WD-xquery-operators-20020816/#" >
        <xf dt-yearMonthDuration>
          P2Y
        </xf dt-yearMonthDuration>
      </EXTENSION>
    <stated-purpose/>
  </RETENTION>
</STATEMENT>
```

Meanwhile, Automart specifies other privacy promises regarding the email information included in the account UPIG – that is, the user email addresses that may be used by its marketing employees (*i.e.* marketing analysts) and by the email flyer server for marketing purposes and that are shared with business partner Automodel (www.automodel.com) upon user consent. The P3P statement specifies these privacy promises as follows:

```
<STATEMENT>
        <EXTENSION>
              <p3p11 STATEMENT-GROUP id="account" />
        </EXTENSION>
        <CONSEQUENCE>We collect essential information to create an account for you and process your order  You can choose to receive
flyers by email</CONSEQUENCE>
        <DATA-GROUP xmlns="http //www w3 org/2002/01/P3Pv1">
      <DATA ref="#user home-info online email"/>
    </DATA-GROUP>
    <PURPOSE>
      <contact required="opt-out"/>
      <EXTENSION>
        <p3ppee PURPOSE-ROLES>
```

```
                <p3ppee marketing required="opt-out">
                        <p3ppee role>MarketingAnalyst</p3ppee role>
                        <p3ppee role>MarketingAdvisor</p3ppee role>
                        <p3ppee role>EmailFlyerServer</p3ppee role>
                </p3ppee marketing>
      </p3ppee PURPOSE-ROLES>
      </EXTENSION>
    </PURPOSE>
    <RECIPIENT>
      <ours>
       <recipient-description>www automart com</recipient-description>
      </ours>
      <same required="opt-out">
       <recipient-description>www automodel com</recipient-description>
      </same>
    </RECIPIENT>
    <RETENTION>
        <EXTENSION xmlns xf="http //www w3 org/TR/2002/WD-xquery-operators-20020816/#" >
          <xf dt-yearMonthDuration>
            P2Y
          </xf dt-yearMonthDuration>
        </EXTENSION>
      <stated-purpose/>
    </RETENTION>
  </STATEMENT>
```

5.1.4.2 The Privacy Practices of the User Private Information Group "profile"

The privacy promises made by Automart regarding the UPI represented by the profile UPIG are comprised of two parts. Firstly, the UPI is stored for two years and is used by the web server of Automart to create online accounts for users and by customer service representatives (CSR) of Automart for communication purposes. Secondly, the UPI collected is optionally used by the marketing employees of Automart for marketing purposes and is shared with business partner Automodel upon user consent. The P3P statement specifies these privacy promises as follows:

```
<STATEMENT>
        <EXTENSION>
                <p3p11 STATEMENT-GROUP id="profile" />
        </EXTENSION>
        <DATA-GROUP xmlns="http //www w3 org/2002/01/P3Pv1">
      <DATA ref="#user name given" />
      <DATA ref="#user name middle" />
      <DATA ref="#user name family" />
      <DATA ref="#user bdate" />
      <DATA ref="#user gender" />
    </DATA-GROUP>

    <PURPOSE>
      <EXTENSION>
        <PPURPOSE xmlns="http //www w3 org/2004/02/P3Pv11">
          <account/>
          <feedback/>
        </PPURPOSE>
      </EXTENSION>
      <current/>
      <contact required="opt-out"/>
```

```
    <EXTENSION>
     <p3ppee PURPOSE-ROLES>
            <p3ppee account>
                <p3ppee role>WebServer</p3ppee role>
            </p3ppee account>
            <p3ppee feedback>
                <p3ppee role>CSR</p3ppee role>
            </p3ppee feedback>
            <p3ppee marketing required="opt-out">
                <p3ppee role>MarketingAnalyst</p3ppee role>
                <p3ppee role>MarketingAdvisor</p3ppee role>
            </p3ppee marketing>
     </p3ppee PURPOSE-ROLES>
    </EXTENSION>
   </PURPOSE>
   <RECIPIENT>
    <ours>
     <recipient-description>www automart com</recipient-description>
    </ours>
    <same required="opt-out">
     <recipient-description>www automodel com</recipient-description>
    </same>
   </RECIPIENT>
   <RETENTION>
    <EXTENSION xmlns xf="http //www w3 org/TR/2002/WD-xquery-operators-20020816/#" >
      <xf dt-yearMonthDuration>
        P2Y
      </xf dt-yearMonthDuration>
     </EXTENSION>
    <stated-purpose/>
   </RETENTION>
  </STATEMENT>
```

5.1.4.3 The Privacy Practices of the User Private Information Group "address"

The privacy promises by Automart regarding the UPI represented by the address UPIG are comprised of two parts. Firstly, the UPI is stored for two years and is used by the web server of Automart to create online accounts and to process payments (*i.e.* is used as billing address), by customer service representatives (CSR) of Automart for communication purposes and by delivery employees (deliveryman) of Automart to ship orders. Secondly, the UPI represented by the address UPIG is used by the marketing employees of Automart for marketing purposes and is shared with business partner Automodel upon user consent. The P3P statement that specifies these privacy promises is as follows:

```
<STATEMENT>
    <EXTENSION>
        <p3p11 STATEMENT-GROUP id="address" />
    </EXTENSION>
    <CONSEQUENCE>You can choose to save additional postal address information with your account thus we can serve you better and
faster  With your consent, we may distribute the data to the specified business partners </CONSEQUENCE>
    <DATA-GROUP xmlns="http //www w3 org/2002/01/P3Pv1">
        <DATA ref="#user home-info postal street"/>
        <DATA ref="#user home-info postal city" />
        <DATA ref="#user home-info postal stateprov" />
```

```
            <DATA ref="#user home-info postal postalcode" />
            <DATA ref="#user home-info postal country" />
      </DATA-GROUP>

    <PURPOSE>
      <EXTENSION>
        <PPURPOSE xmlns="http //www w3 org/2004/02/P3Pv11">
          <account/>
          <payment/>
          <feedback/>
          <delivery/>
        </PPURPOSE>
      </EXTENSION>
      <contact required="opt-out"/>
      <current/>
      <EXTENSION>
       <p3ppee PURPOSE-ROLES>
            <p3ppee account>
                  <p3ppee role>WebServer</p3ppee role>
            </p3ppee account>
            <p3ppee payment>
                  <p3ppee role>WebServer</p3ppee role>
            </p3ppee payment>
            <p3ppee delivery>
                  <p3ppee role>deliveryman</p3ppee role>
            </p3ppee delivery>
            <p3ppee feedback>
                  <p3ppee role>CSR</p3ppee role>
            </p3ppee feedback>
            <p3ppee marketing required="opt-out">
                  <p3ppee role>MarketingAnalyst</p3ppee role>
                  <p3ppee role>MarketingAdvisor</p3ppee role>
                  <p3ppee role>FlyerDistributor</p3ppee role>
            </p3ppee marketing>
       </p3ppee PURPOSE-ROLES>
      </EXTENSION>
    </PURPOSE>
    <RECIPIENT>
      <ours>
       <recipient-description>www automart com</recipient-description>
      </ours>
      <same required="opt-out">
       <recipient-description>www automodel com</recipient-description>
      </same>
    </RECIPIENT>
    <RETENTION>
      <EXTENSION xmlns xf="http //www w3 org/TR/2002/WD-xquery-operators-20020816/#" >
          <xf dt-yearMonthDuration>
            P2Y
          </xf dt-yearMonthDuration>
       </EXTENSION>
      <stated-purpose/>
    </RETENTION>
  </STATEMENT>
```

5.1.4.4 The Privacy Practices of the User Private Information Group "payment"

The privacy promises by Automart regarding the UPI represented by the payment UPIG are comprised of two parts. Firstly, the UPI is stored for two years and is used by the web server of Automart to create online accounts for users and to process orders and payments. Secondly, the UPI represented by the payment UPIG is used by marketing employees of Automart for marketing purposes (*e.g.* promoting credit products) and is shared with business partner Automodel upon user consent. The P3P statement that specifies these privacy promises is as follows:

```
<STATEMENT>
        <EXTENSION>
                <p3p11 STATEMENT-GROUP id="payment" />
        </EXTENSION>

        <DATA-GROUP xmlns="http //www w3 org/2002/01/P3Pv1">
                <DATA ref="#user payment creditcard cctype" />
                <DATA ref="#user payment creditcard ccnum" />
                <DATA ref="#user payment creditcard holdername" />
                <DATA ref="#user payment creditcard expiry" />
                <DATA ref="#user payment creditcard securitycode" />
        </DATA-GROUP>
      <PURPOSE>
        <EXTENSION>
         <PPURPOSE xmlns="http //www w3 org/2004/02/P3Pv11">
           <sales/>
           <account/>
           <payment/>
           <marketing/>
         </PPURPOSE>
        </EXTENSION>
        <current/>
        <EXTENSION>
         <p3ppee PURPOSE-ROLES>
                <p3ppee account>
                        <p3ppee role>WebServer</p3ppee role>
                </p3ppee account>
                <p3ppee sales>
                        <p3ppee role>WebServer</p3ppee role>
                </p3ppee sales>
                <p3ppee payment>
                        <p3ppee role>WebServer</p3ppee role>
                </p3ppee payment>
                <p3ppee marketing required="opt-out">
                        <p3ppee role>MarketingAnalyst</p3ppee role>
                        <p3ppee role>MarketingAdvisor</p3ppee role>
                </p3ppee marketing>
        </p3ppee PURPOSE-ROLES>
        </EXTENSION>
      </PURPOSE>
      <RECIPIENT>
        <ours>
         <recipient-description>www automart com</recipient-description>
        </ours>
        <same required="opt-in">
         <recipient-description>www automodel com</recipient-description>
```

```
    </same>
  </RECIPIENT>
  <RETENTION>
    <EXTENSION xmlns xf="http //www w3 org/TR/2002/WD-xquery-operators-20020816/#" >
        <xf dt-yearMonthDuration>
          P2Y
        </xf dt-yearMonthDuration>
      </EXTENSION>
    <stated-purpose/>
  </RETENTION>
</STATEMENT>
```

5.1.4.5 Database Compliance with EAPEX

On the registration page of Automart, there is a hidden input field for each UPIG. Each hidden input field is used by the EAPEX user agent to submit the generated UPCP rule. In the UPI and UPP storage (*i.e.* database) of Automart, the schemas of the tables used to store relevant UPI pieces define the four UPIGs except for the primary keys and the additional attributes for storing the UPCP rules. The schemas of the tables in the UPI and UPP storage are shown in the diagram below. Note that all the names of the attributes except the primary keys and the ones for storing UPCP rules are created to be consistent with the corresponding P3P user data type names in order to generate effective resource XPath expressions when transforming the P3P policies into the corresponding OPCPs. For example, **profile(given,...)** is consistent with the P3P <STATEMENT-GROUP id="**profile**"> and the <DATA ref="#user.name.given"/> elements, so that the resource XPath expression "CustomerRecord/**profile/given**" generated during the transformation is accurate and effective.

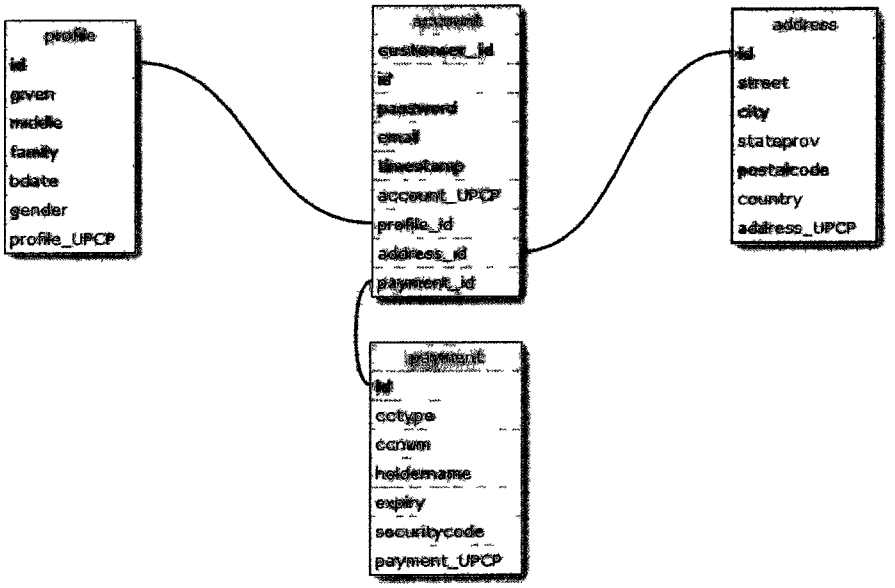


Diagram 5.1.4.5 – Database Schema of Automart

5.1.5 Automart's Privacy Policy in Human Readable Format (Natural Language)

Our Privacy Commitment

At Automart, we care about your privacy. Without your consent, we will never share your credit card information or any other aspect of your private information with any third party or use it for marketing purpose. Only with your permission will we share information with carefully selected business partners in accordance with equitable privacy practices. At Automart, EAPEX is deployed so that your privacy preferences are collected and stored together with your private information and are actually used to manage access to your private information in order to enforce our promises with respect to your privacy preferences.

Automart is a licensee of the PPESeal (Privacy Protection Enforcement Seal) Program. The PPESeal program ensures your privacy by holding website licensees to high privacy standards and by confirming with independent auditors that these privacy practices are being followed properly.

What We Collect and Why

When you visit and browse through our online store, we collect:

- The basic information about your user agent (*i.e.* your internet browser) and connection (*i.e.* IP address) is collected to make sure that we can provide you with the proper customization and is used for security purposes;
- Aggregated information on what pages visitors browse most often to improve our site.

If you choose to purchase a used vehicle at Automart we will ask you for more information including:

- A login user name, password and email which are used to create an account so that you can update your information at any time in the future or reset your password via email in case of password loss;
- Credit card information to complete your purchase;
- Your name and postal address, so that we can have your purchase delivered to you and contact you in the future;
- Other demographic information (e.g. birthday and gender) to enable us to tailor our services to you.

How to Set Your Privacy Preferences

Automart is an EAPEX enabled website. You can set your privacy preferences using any EAPEX user agent.

Changing and Updating Private Information

Registered customers can change or update all their private information and privacy preferences by going to the account section of Automart at http://www.automart.com/profile.jsp. You can also choose to update the demographic information, postal address or credit card information stored in your online account or delete your account at any time.

Cookies

Automart never uses cookies to store identifiable data.

Data Retention

We will keep the information about you and your purchases for two years following the date of registration.

Contacting Automart

Questions regarding this statement should be directed to:

Automart
1680 Bank Street
Ottawa, ON K3C6A8 Canada
Email: hudianshu@automart.com
Telephone: 1 613-888-8888

If we have not responded to your inquiry or your inquiry has not been satisfactorily addressed, you can contact PPESeal at http://www.PPESeal.org/privacyseal. Automart will correct all errors or inappropriate actions arising from or connected with the privacy policy.

5.2 Privacy Controller

5.2.1 Overview

The Privacy Controller is a fully functional EAPEX user agent implemented as an extension of Firefox to address two major types of privacy violation: 1) the dissemination of users' private data to third parties and 2) the marketing uses of users' private data by legitimate receivers without users' consent. It provides a user-friendly GUI that allows a naive user to easily store his/her private information and privacy preferences. When a user browses an EAPEX website with the Privacy Controller, it automatically parses the privacy policies written in P3P with P3PPEE and finds the one that covers the current page being browsed. Once it detects UPI collection by checking the relevance of the policy, it automatically checks the marketing usage and third-party recipients against the user's privacy preferences. If there is a contradiction between the policy and user preferences, the Privacy Controller automatically generates UPCP

rules that deny the contradicting privacy practices and fill them in the html form together with the required UPI without user intervention. In addition, the user can easily use the Privacy Controller to generate a valid APPEL privacy preference file based on his current privacy preference settings. This generated APPEL file can be used by the Privacy Controller to import user privacy preferences and by any other APPEL enabled user agent.

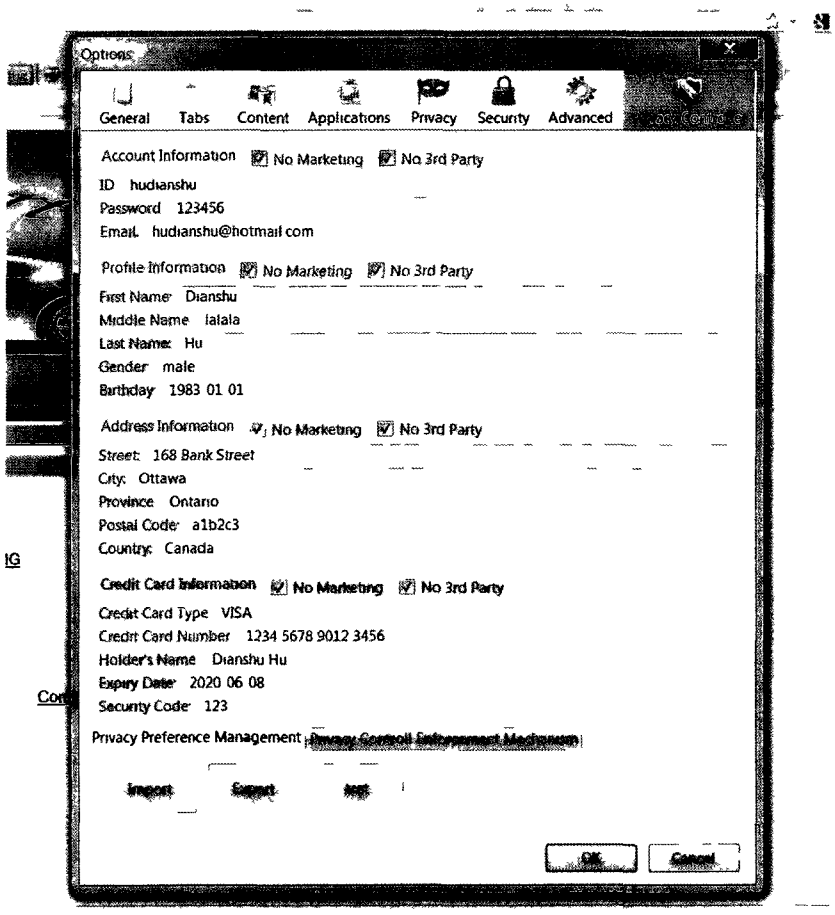
5.2.2 Technologies Employed and GUI Design

5.2.2.1 Technologies Employed

- IDE: NetBeans 6.9.1
- Programming language: JavaScript
- IDE plug-in for developing Firefox extension: Foxbean
- XML processing: DOMParser
- GUI markup language: XUL (the XML-based User-interface Language)
- Mechanism to handle I/O: XPCOM (Cross-Platform Component Object Model) JavaScript lacks a function for handling I/O and file operations and XPCOM is used to handle the operations. In Internet Explorer, ActiveX is used instead.

5.2.2.2 GUI Design

The GUI is implemented by using XUL which is exclusively employed in Mozilla applications like Firefox and Thunderbird and their extensions. The screen shot below shows the GUI of Privacy Controller.



The GUI allows a user to enter various UPI pieces that are usually required by e-businesses. Note that the UPIGs defined by Automart are used to aggregate the relevant UPI. The grouping can be dynamically constructed since the UPIGs of an EAPEX enabled website are included in its P3P privacy policy. Due to the reason that Automart is the only EAPEX website currently being implemented, its UPIGs are used directly. The user can prevent marketing usage of his/her UPI in each UPIG by the first-party recipient by checking the corresponding "No Marketing" checkbox. Similarly, third-party sharing of the UPI in each UPIG can be prevented by checking the corresponding "No 3rd Party" checkbox. This GUI is very user-friendly so that even a user who has no idea at all about the technologies involved can easily adopt it. The input fields of the GUI are bound to the preferences system by using the <preferences> element in the XUL file of the GUI. For example, within the <preferences> tag shown in the code segment below, a preference bound to the "extensions.PrivacyController.id" preference string is defined.

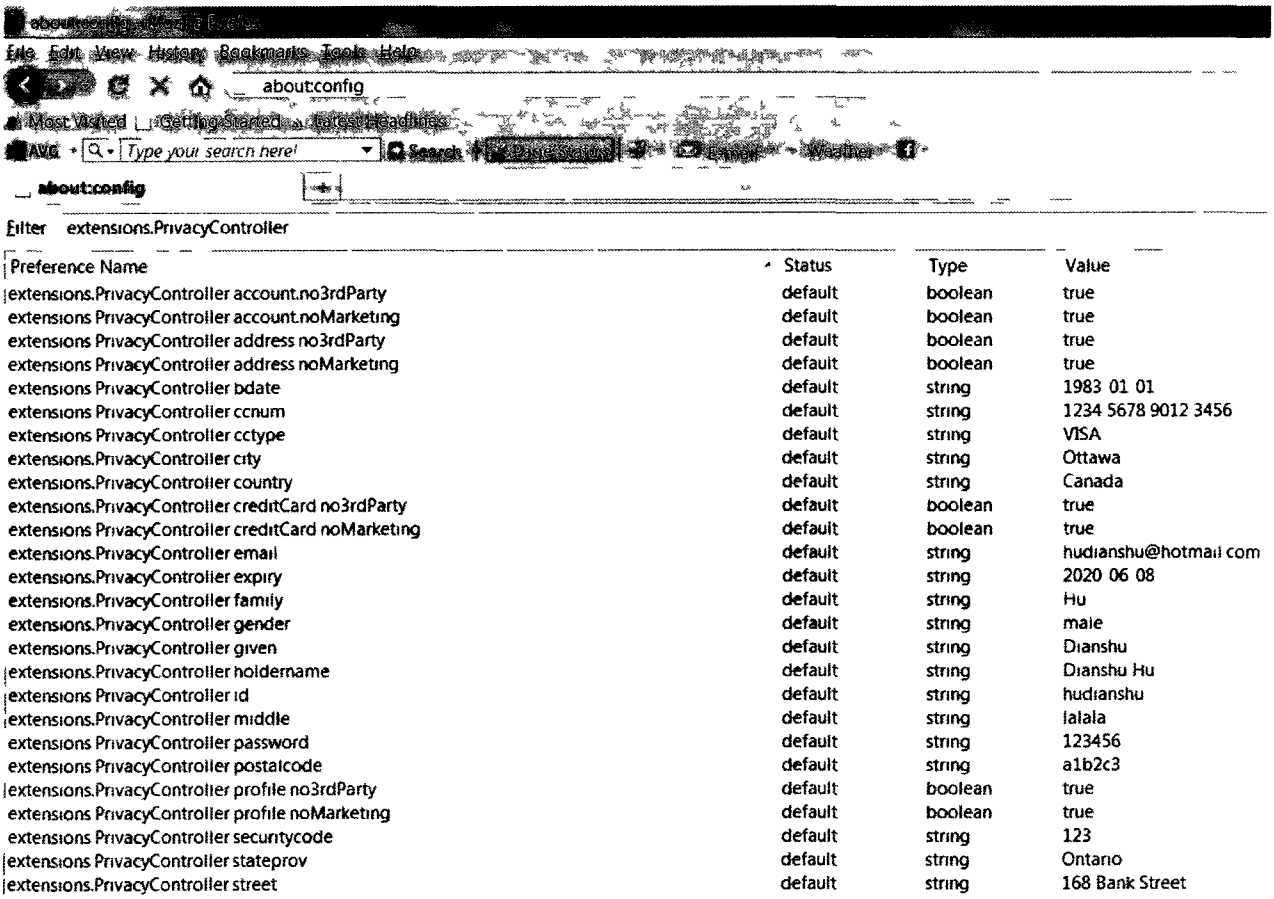
```
<preferences>
  <preference id="idPref" name="extensions.PrivacyController.id" type="string" />
  ...
```

</preferences>

Then it is associated with the ID textbox by the following line of XUL code:

<textbox id="idField" flex="1" preference="**idPref**"/>

The screen shot below shows all the preferences (*i e* the UPI and the privacy preferences of UPIGs) used by the Privacy Controller that are stored using the preferences system of Firefox.



Preference Name	Status	Type	Value
extensions.PrivacyController account.no3rdParty	default	boolean	true
extensions.PrivacyController account.noMarketing	default	boolean	true
extensions.PrivacyController address.no3rdParty	default	boolean	true
extensions.PrivacyController address.noMarketing	default	boolean	true
extensions.PrivacyController bdate	default	string	1983 01 01
extensions.PrivacyController ccnum	default	string	1234 5678 9012 3456
extensions.PrivacyController cctype	default	string	VISA
extensions.PrivacyController city	default	string	Ottawa
extensions.PrivacyController country	default	string	Canada
extensions.PrivacyController creditCard.no3rdParty	default	boolean	true
extensions.PrivacyController creditCard.noMarketing	default	boolean	true
extensions.PrivacyController email	default	string	hudianshu@hotmail.com
extensions.PrivacyController expiry	default	string	2020 06 08
extensions.PrivacyController family	default	string	Hu
extensions.PrivacyController gender	default	string	male
extensions.PrivacyController given	default	string	Dianshu
extensions.PrivacyController holdername	default	string	Dianshu Hu
extensions.PrivacyController id	default	string	hudianshu
extensions.PrivacyController middle	default	string	lalala
extensions.PrivacyController password	default	string	123456
extensions.PrivacyController postalcode	default	string	a1b2c3
extensions.PrivacyController profile.no3rdParty	default	boolean	true
extensions.PrivacyController profile.noMarketing	default	boolean	true
extensions.PrivacyController securitycode	default	string	123
extensions.PrivacyController stateprov	default	string	Ontario
extensions.PrivacyController street	default	string	168 Bank Street

5.2.3 Key Functions and Code Segments of the Privacy Controller

In this section, the major functions and code segments that make the Privacy Controller function are listed and explained.

5.2.3.1 Intercepting Page Load

In order to intercept the page load and parse the P3P policy reference file and P3P privacy policies, a new page load event handler is added to the tabbed browser of Firefox. The function adding this handler is shown below

```
function addPageLoadHandler(event){
  gBrowser.addEventListener("load", privacyControlEnforcement, true);
}
```

5.2.3.2 Parsing the P3P Policy Reference File and the P3P Privacy Policies

After intercepting the page load, the code segment below is invoked to parse the P3P reference file of the website being visited. Similarly, the P3P privacy policies specified in the P3P reference file are parsed.

```
if (window.XMLHttpRequest){
        httpRequest=new XMLHttpRequest();

}else {
        alert("XMLHttpRequest is not working");
    }
httpRequest.open("GET",websiteURL+"/p3p/p3p.xml",false)
httpRequest.send(null);
// refXMLDoc is the DOM object of the P3P reference file parsed
var refXMLDoc=httpRequest.responseXML;
```

5.2.3.3 Accessing the P3P/P3PPEE Elements of the Parsed P3P Files

The code segment below shows how to access nodes of a certain name (*e.g.* "POLICY"). Note that the "policyEle" is the root element of the DOM object "policyXMLDoc".

```
var policyEle = policyXMLDoc.getElementsByTagNameNS(p3p,"POLICY")[0];
```

5.2.3.4 Generating XACML Rule Elements

The code segment shown below generates a DOM object with the root element "Rule" and appends a child element "Target".

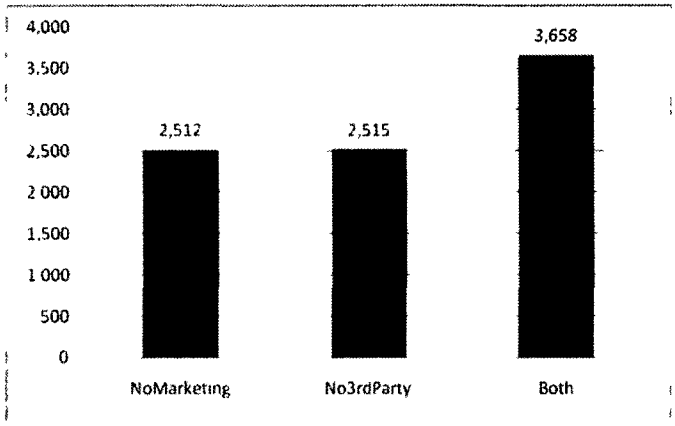
```
var ruleDoc = document.implementation.createDocument(xacml, "Rule", null);
var ruleEle = ruleDoc.documentElement;
```

```
var targetEle = ruleDoc.createElementNS(xacml,"Target");
ruleEle.appendChild(targetEle);
```

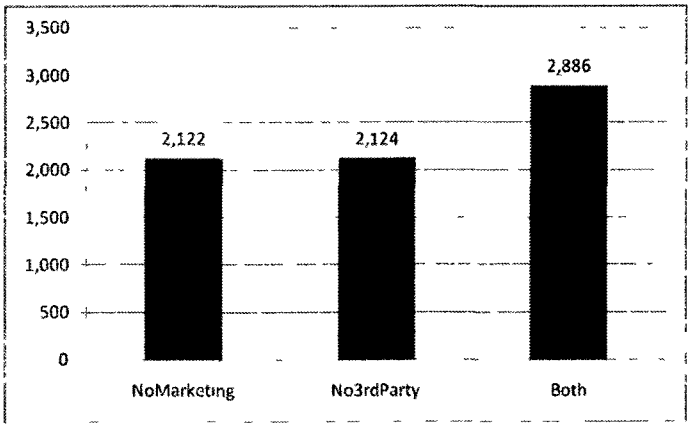
5.2.4 Testing the Bandwidth Overhead Incurred by Employing EAPEX (Automart and Privacy Controller)

The sizes (in bytes) of different UPCP rule outputs generated by setting various privacy preference combinations (*i.e.* "No Marketing" only, "No 3rd Party" only or both) for each UPIG are measured and compared. They represent the additional bandwidth usage incurred by applying the negotiation mechanism to each UPIG. The total bandwidth overhead incurred to enforce EAPEX is comprised of the total size of the UPIG rules generated for the UPIGs plus the size of the P3P files parsed.

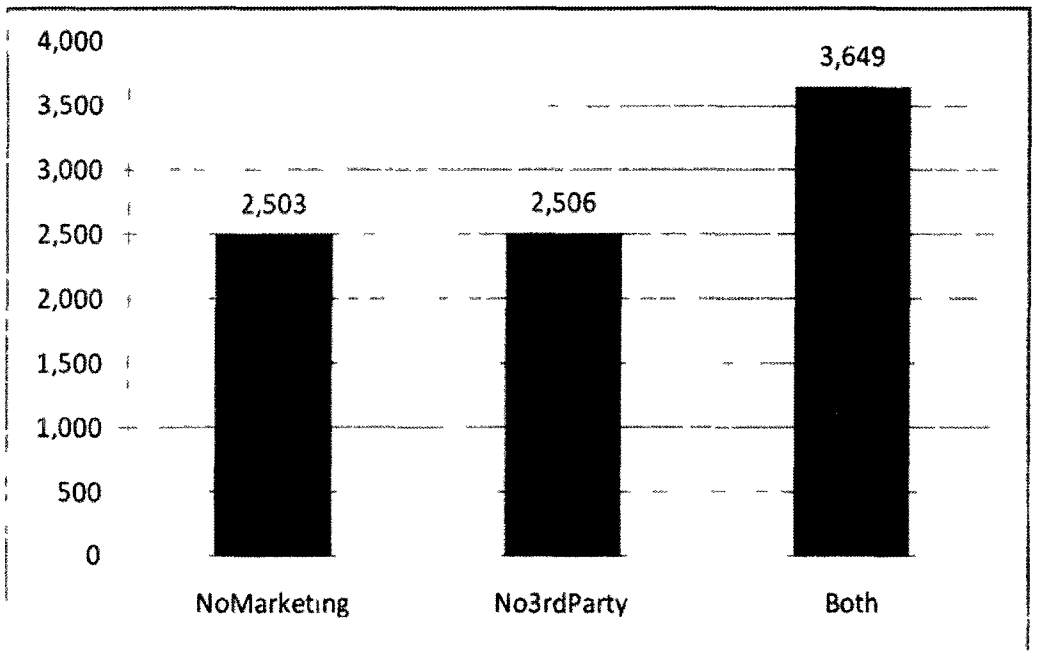
5.2.4.1 Overhead by the User Private Information Group "account"



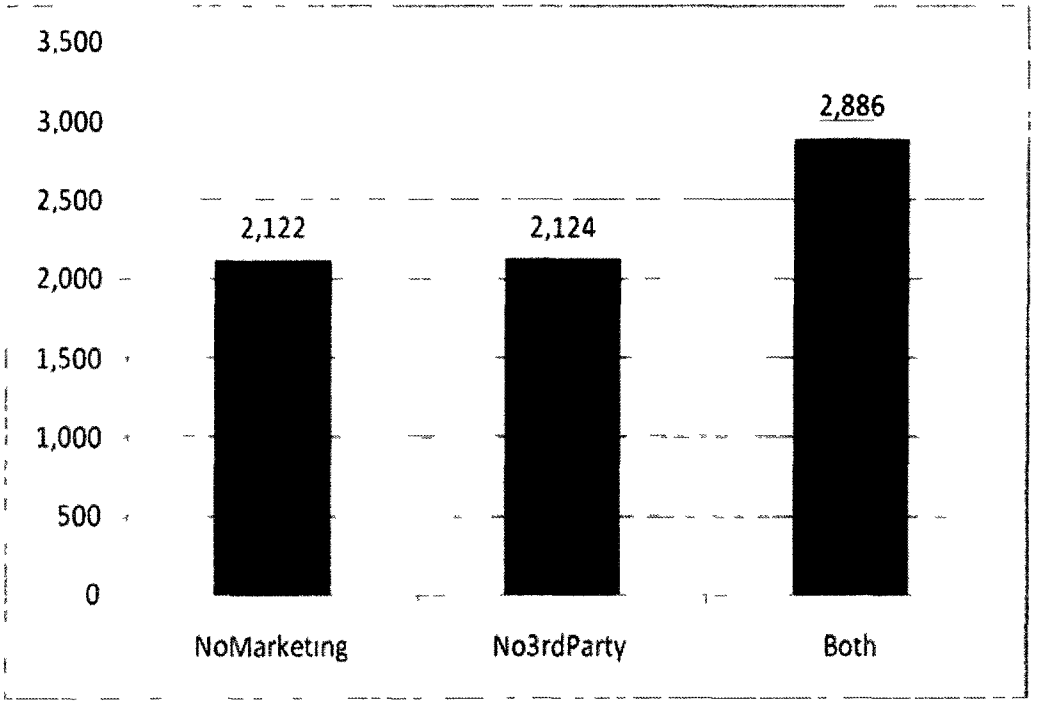
5.2.4.2 Overhead by the User Private Information Group "profile"



5.2.4.3 Overhead by the User Private Information Group "address"



5.2.4.4 Overhead by the User Private Information Group "payment"



5.2.4.5 Total Bandwidth Overhead Incurred by EAPEX (Automart and Privacy Controller)

The total size of the P3P files deployed at Automart is 55,895 bytes. The total size of UPCP rules generated by the Privacy Controller in the worst case scenario (*i.e.* when both "No Marketing" and "No 3rd Party" are checked for every UPIG) is 13,079 bytes. Therefore, the total Bandwidth Overhead Incurred by EAPEX (Automart and Privacy Controller) in the worst case scenario is 68,974 bytes (*i.e.* 67.36 KB) which is not significant at all.

5.2.5 Testing the Time for Parsing WPP and for Negotiation (Automart and Privacy Controller)

The time for Privacy Controller with increasingly strict preference settings to generate UPCP rules (*i.e.* negotiate privacy control practices) is measured. The least strict preference setting allows Automart to use any piece of UPI for marketing purposes and shares the information with the third party Automodel, while the most strict preference setting prevents marketing usage and third-party sharing of any UPI. The average time for Privacy Controller with various preference settings to negotiate privacy practices is 4.19 milliseconds. The average time for negotiation with the most strict preference setting is 5.75 milliseconds. The time for Privacy Controller to parse the WPP of Automart is also measured. It takes 15.93 milliseconds on average. The total time overhead incurred by the negotiation mechanism in the worst case scenario which is comprised of the longest time period for parsing the WPP plus the longest time period for Privacy Controller with the most strict preference setting to negotiate is 40 milliseconds (*i.e.* 0.04 second). Because the interruption caused by deploying EAPEX is instantaneous, web users will not therefore suffer any inconvenience as a result of using EAPEX enabled e-businesses. It should be noted that the validity of the test result for the time needed for parsing the WPP of Automart is limited since the tests were conducted on a local system. However, the web users should not suffer from the interruption for parsing the WPP since most of them already have a broadband internet connection.

6 Conclusions and Future Work

The major objective of this thesis is to enhance the Architecture for Privacy Enforcement using XML (APEX) to make it practical. To achieve this objective, the Privacy Enforcement Extension for P3P 1.1 is proposed through this thesis research. The mapping of privacy promises specified in P3P with P3PPEE to the corresponding privacy control practices specified in XACML with the privacy profile extension is proposed. In addition, a negotiation mechanism is developed for reconciling user privacy preferences and organizational privacy promises in order to address users' privacy concerns. Furthermore, a verification mechanism that can be used by external auditors to verify e-businesses' compliance with EAPEX is proposed. Finally, both the EAPEX enabled website "Automart" and the EAPEX user agent "Privacy Controller" in an e-business setting are implemented. Moreover, both the bandwidth overhead and the time overhead incurred by deploying EAPEX are measured. Based on the test results, the web users' privacy concerns can be addressed without any inconvenience resulting from the use of EAPEX enabled e-businesses. The contributions of this thesis also include proposing a theoretical architecture design of the Architecture for Privacy Enforcement using PBE (APEP). APEP seems to be less complex than EAPEX because the PBE approach instead of the combination of XACML and SSL is used to achieve privacy enforcement. In general, both architecture designs proposed in this thesis research achieve privacy enforcement in an e-business environment. APEP may be more suitable for small e-businesses while EAPEX seems to be more appropriate in a large e-business setting. The uses of EAPEX and APEP can span other environments where users' privacy preferences must be enforced.

Although the negotiation mechanism is implemented and the bandwidth and time overheads are measured, the overall effectiveness of EAPEX in terms of privacy enforcement is not tested because the automated transformation mechanism, which automatically transforms the privacy policies specified in P3P with P3PPEE into the corresponding OPCPs, and the Privacy Enforcement Model, which enforces the negotiated privacy control practices, are not implemented and tested. One research area that could extend the work presented in this thesis is to implement the remaining parts of EAPEX and to test its overall effectiveness. Another possible research area would be to implement APEP so that its performance and effectiveness can be compared with those of EAPEX.

Reference

[1] K. Barbieri. "Architecture for Privacy Enforcement using XML (APEX)" Master Project Report. Supervisor: Dr. Carlisle Adams, University of Ottawa. August 2004.

[2] P. Hop-Tindall. "Privacy Impact Assessment - Obligation or Opportunity: The Choice is Ours", presented at CSE ITS 2002, Ottawa, Canada. http://www.dataprivacy.com/mod/fileman/files/PIA_Material.pdf

[3] OASIS eXtensible Access Control Markup Language (XACML), Version 1.0, Committee Specification, 18 February 2003. Location: http://www.oasis-open.org/committees/download.php/2406/oasis-xacml-1.0.pdf

[4] G. Yee, and L. Korba. "Semi-Automated Derivation of Personal Privacy Policies" In Proceedings of the 2004 Information Resources Management Association International Conference (IRMA 2004). New Orleans, Louisiana, USA. May 23-26, 2004. NRC 46539

[5] L. F. Cranor. "Web Privacy with P3P" Published by O'Reilly & Associations, Inc. September 2002. First Edition. ISBN: 0-596-00371-4

[6] WWW Consortium, "The Platform for Privacy Preferences 1.0 (P3P 1.0) Specification" W3C Recommendation, 16 April 2002, http://www.w3.org/TR/2002/REC-P3P-20020416/.

[7] P. Ashley, S. Hada, G. Karjoth, C. Powers and M. Schunter. "Enterprise Privacy Authorization Language (EPAL 1.2)", W3C Member Submission 10 November 2003, http://www.w3.org/submission/2003/SUBM-EPAL-20031110/.

[8] S. Lakshminarayanan, R. Ramamoorthy, and P. C. Hung. "Conflicts in Inter-prise EPAL Policies" Presented at W3C Workshop on the Future of P3P. June 19-20, 2003, Kiel (Schleswig-Holstein, Germany).

[9] G. Karjoth, M. Schunter, and E. Van Herreweghen. "Translating Privacy Practices into Privacy Promises - How to Promise What You Can Keep" In Proceedings of the 4th International Workshop on Policies for Distributed Systems and Networks (POLICY '03).

[10] D. Li and H. Yan. "Automated Translations for Architecture for Privacy enforcement using XML (APEX)" Honors Project Report. Supervisor: Dr. Carlisle Adams, University of Ottawa. April 2004.

[11] "Sun's XACML Implementation Programmer's Guide for Version 1.1" http://sunxacml.sourceforge.net/guide.html. Last Updated: November 5, 2003.

[12] WWW Consortium. "P3P 1.0 Implementations" http://www.w3.org/p3p/implementations.html.

[13] Y. Beres, P. Bramhall, M.Casassa Mont, M. Gittler, and S. Person. "Accountability and Enforceability of Enterprise Privacy Policies" Trusted Systems Laboratory (TSL), Hewlett-Packard Laboratories, HPL-2003-119, 2003.

[14] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu. "Hippocratic Databases" In Proceedings of 28th VLDB Conference, Hong Kong, China, 2002.

[15] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu. "Implementing P3P Using Database Technologies" In Proceedings of the 19th International Conference on Data Engineering, Bangalore, India. March 2003.

[16] M. Backes, W. Bagga, G. Karjoth, and M. Schunter. "Efficient comparison of Enterprise Privacy Policies" In Proceedings of the 2004 ACM Symposium on Applied Computing (SAC '04). Nicosia, Cyprus. March 14-17, 2004.

[17] M. Langheinrich. "A Privacy Awareness System for Ubiquitous Computing Environments" Ubicomp 2002.

[18] A. Felty, and S. Matwin. "Privacy-Oriented Data Mining by Proof Checking" in Sixth European Conference on Principles of Data Mining and Knowledge Discovery, Springer-Verlag, LNCS 2431, August 2002.

[19] P. Ashley, S. Hada, G. Karjoth, C. Powers and M. Schunter. "From privacy promises to privacy management: a new approach for enforcing privacy throughout an enterprise". Proceedings of 2002 In New Security Paradigms Workshop, Virginia Beach, Virginia, 2002, ISBN: 1-58113-598-X.

[20] M. Mahmoudian. "Developing an Internet Access Control Privacy for a Web Site using an Automated Privacy Policy Mapping" Honor Project Report. Supervisor: Dr. Carlisle Adams, University of Ottawa. December 2004.

[21] C. Adams and P. Madsen, "Privacy and XML", O'Reilly XML.COM, May 01, 2002, available at http://www.xml.com/pub/a/2002/04/17/privacy.html/.

[22] P. C. K Hung, E. Ferrari, B. Carminati. "Towards standardized Web services privacy technologies", IEEE International Conference on Web Services, 2004, Proceedings, July 6-9 2004, Pages: 174-181.

[23] AT&T Privacy Bird. http://www.privacybird.org

[24] WWW Consortium, "The Platform for Privacy Preferences 1.1 (P3P 1.1) Specification" W3C Working Group Note, 13 November 2006, http://www.w3.org/TR/2006/NOTE-P3P11-20061113/.

[25] WWW Consortium, "A P3P Preference Exchange Language 1.0 (APPEL 1.0)" W3C Working Draft, 15 April 2002, http://www.w3.org/TR/P3P-preferences/.

[26] OASIS eXtensible Access Control Markup Language (XACML), Version 1.1, Committee Specification, 07 August 2003. Location: http://www.oasis-open.org/committees/xacml/repository/cs-xacml-specification-1.1.pdf

[27] OASIS eXtensible Access Control Markup Language (XACML), Version 2.0, Committee Specification, 01 Feb 2005. Location: http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-spec-os.pdf

[28] OASIS Hierarchical resource profile of XACML v2.0, Committee Standard, 01 Feb 2005. Location: http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-hier-profile-spec-os.pdf

[29] OASIS Multiple resource profile of XACML v2.0, Committee Standard, 01 Feb 2005. Location: http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-mult-profile-spec-os.pdf

[30] OASIS Privacy policy profile of XACML v2.0, Committee Standard, 01 Feb 2005. Location: http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-privacy_profile-spec-os.pdf

[31] Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data (1995) http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31995L0046:EN:HTML

[32] Privacy and Personal Information Protection Act (1998) http://www.lawlink.nsw.gov.au/lawlink/privacynsw/ll_pnsw.nsf/pages/PNSW_03_ppipact

[33] Personal Information Protection and Electronic Documents Act (PIPEDA 2000) http://www.priv.gc.ca/leg_c/leg_c_p_e.cfm

[34] "Web Privacy with P3P" by Lorrie Faith Cranor, O'Reily & Associates 2002

[35] JRC Policy Workbench
http://jrc-policy-api.sourceforge.net/

[36] WWW Consortim, "Document Object Model"
http://www.w3.org/DOM/DOMTR

[37] Yee, George (2006). Protecting Privacy Using XML, XACML, and SAML, Privacy Protection for E-Services (pp. 203-232).Hershey: Idea Group Publishing.

[38] http://www.truste.com

[39] W. Bagga and R. Molva, "Policy-Based Cryptography and Applications", in lecture Notes in Computer Science, pp-72-87, Springer Berlin / Heidelberg, 2005.

[40] Personal communications from Professor Carlisle Adams to the author

[41] K. Garson and C. Adams. "Security and Privacy System Architecture for an e-Hospital Environment" In IDtrust 08 Proceedings of the 7th symposium on Identity and trust on the Internet, Gaithersburg, MD, USA. March 4-6, 2008.

[42] A. F. Westin, "Privacy and freedom" (Fifth ed.). New York: Atheneum, 1967, page 7.

[43] G. Karjoth, M. Schunter, and M. Waidner. "Platform for enterprise privacy practices: Privacy-enabled management of customer data" In 2nd International Workshop on Privacy Enhancing Technologies (PET2002), volume 2482 of Lecture Notes in Computer Science, pages 69-84. Springer-Verlag, April 2002.

[44] E. SHERMAN. Privacy policies are great—for phds, September 4, 2008.
http://industry.bnet.com/technology/1000391/privacypolicies-are-great-for-phds/

[45] S. Egelman, J. Tsai, L. F. Cranor, and A. Acquist. "Timing is everything? The Effects of Timing and Placement of Online Privacy Indicators." In CHI 09 Proceedings of the 27th international conference on Human Factors in Computing Systems, pages 319-328, Boston, Massachusetts, USA. April 4-9, 2009.

[46] I. Reay, S. Dick, and J. Miller. "A Large-Scale Empirical Study of P3P Privacy Policies: Stated Actions vs. Legal Obligations." ACM Transactions on the Web (TWEB), volume 3, issue 2, article 6 ACM, New York, NY, USA. April, 2009.

[47] S. Vimercati, S. Foresti, S, Jajodia, S. Paraboschi, and P. Samarati. "Encryption Policies for Regulating Access to Outsourced Data." ACM Transactions on Database System (TODS), volume 35, issue 2, article 12 ACM, New York, NY, USA. April 2010.

[48] R. Bobba, H. Khurana, AlTurki, and F. Ashraf. "PBES: A Policy Based Encryption System with Application to Data Sharing in the Power Grid." In ASIACCS 09 Proceedings of the 4th international Symposium on Information, Computer, and Communications Security, Sydney, NSW, Australia. March 10-12, 2009.

Appendix 1: The Privacy Enforcement Extension for P3P 1.1

```
<?xml version="1.0" encoding="UTF-8"?>
<schema elementFormDefault="qualified" targetNamespace="http://www.example.org/P3PPEE"
 xmlns="http://www.w3.org/2001/XMLSchema"
 xmlns:p3p="http://www.w3.org/2002/01/P3Pv1"
 xmlns:p3p11="http://www.w3.org/2006/01/P3Pv11"
 xmlns:p3p11bds="http://www.w3.org/2006/01/P3Pv11BDS"
 xmlns:p3ppee="http://www.example.org/P3PPEE">

 <import namespace="http://www.w3.org/XML/1998/namespace" schemaLocation="http://www.w3.org/2001/xml.xsd"/>
 <import namespace="http://www.w3.org/2002/01/P3Pv1" schemaLocation="http://www.w3.org/2002/01/P3Pv1.xsd"/>
 <import namespace="http://www.w3.org/2006/01/P3Pv11" schemaLocation="http://www.w3.org/2006/01/P3Pv11.xsd" />
 <import namespace="http://www.w3.org/2006/01/P3Pv11BDS"
schemaLocation="http://www.w3.org/2006/01/P3Pv11BDS.xsd" />

 <element name="datatype" type="p3ppee:datadefComplexType" />

 <complexType name="datadefComplexType">
        <all>
        <element minOccurs="0" name="dynamic" type="p3p11bds:dynamicComplexType" />
        <!-- ********** Substitue ********** -->
                        <element minOccurs="0" name="user" type="p3ppee:userComplexType" />
                <element minOccurs="0" name="thirdparty" type="p3p11bds:thirdpartyComplexType" />
                <element minOccurs="0" name="business" type="p3p11bds:businessComplexType" />
        </all>
   <attribute type="p3p:yes_no" default="no" use="optional" name="optional" />
   </complexType>

 <!-- ********** Base Data Schema Extension ********** -->
 <complexType name="userComplexType">
        <complexContent>
                <extension base="p3p11bds:userComplexType">
                        <all>
                                <element minOccurs="0" name="payment" type="p3ppee:paymentComplexType">
                        <annotation>
                                <appinfo>
                                                        <CATEGORIES
xmlns="http://www.w3.org/2002/01/P3Pv1">
                                        <financial />
                                </CATEGORIES>
                                </appinfo>
                                <documentation>User's Payment Information</documentation>
                        </annotation>
                        </element>
                </all>
                </extension>
        </complexContent>
 </complexType>

 <complexType name="paymentComplexType">
        <all>
                <element minOccurs="0" name="creditcard" type="p3ppee:creditcardComplexType">
```

```
        <annotation>
                <appinfo>
                                        <CATEGORIES xmlns="http://www.w3.org/2002/01/P3Pv1">
                        <financial />
                </CATEGORIES>
                </appinfo>
                <documentation>User's Credit Card Payment Information</documentation>
        </annotation>
        </element>
        </all>
</complexType>

<complexType name="creditcardComplexType">
        <all>
                <element minOccurs="0" name="cctype">
                        <annotation>
                <appinfo>
                <CATEGORIES xmlns="http://www.w3.org/2002/01/P3Pv1">
                        <financial />
                </CATEGORIES>
                </appinfo>
                <documentation>Credit Card Type</documentation>
        </annotation>
                </element>

                <element minOccurs="0" name="ccnum">
                        <annotation>
                <appinfo>
                <CATEGORIES xmlns="http://www.w3.org/2002/01/P3Pv1">
                        <financial />
                </CATEGORIES>
                </appinfo>
                <documentation>Credit Card Number</documentation>
        </annotation>
                </element>

                <element minOccurs="0" name="holdername">
                        <annotation>
                <appinfo>
                <CATEGORIES xmlns="http://www.w3.org/2002/01/P3Pv1">
                        <financial />
                </CATEGORIES>
                </appinfo>
                <documentation>Credit Card Holder's Name</documentation>
        </annotation>
                </element>

                <element minOccurs="0" name="expiry">
                        <annotation>
                <appinfo>
                <CATEGORIES xmlns="http://www.w3.org/2002/01/P3Pv1">
                        <financial />
                </CATEGORIES>
                </appinfo>
```

```
                <documentation>Credit Card Expiration Date</documentation>
        </annotation>
                </element>

                <element minOccurs="0" name="securitycode">
                        <annotation>
                <appinfo>
                <CATEGORIES xmlns="http://www.w3.org/2002/01/P3Pv1">
                        <financial />
                </CATEGORIES>
                </appinfo>
                <documentation>Credit Card Security Code</documentation>
        </annotation>
                </element>
        </all>
</complexType>

<!-- ********** Conciliate PURPOSE and PPURPOSE, and adding contained roles********** -->
 <element name="PURPOSE-ROLES">
 <complexType>
  <sequence>
          <choice maxOccurs="unbounded">
                <!-- ********** purposes from PURPOSE element********** -->
                 <element name='admin' type="p3ppee:purpose-role-value"/>
         <element name='develop' type="p3ppee:purpose-role-value"/>
         <element name='tailoring' type="p3ppee:purpose-role-value"/>
         <element name='pseudo-analysis' type="p3ppee:purpose-role-value"/>
         <element name='pseudo-decision' type="p3ppee:purpose-role-value"/>
         <element name='individual-analysis' type="p3ppee:purpose-role-value"/>
         <element name='individual-decision' type="p3ppee:purpose-role-value"/>
         <element name='historical' type="p3ppee:purpose-role-value"/>

         <!-- ********** purposes from PPURPOSE element********** -->
    <element name="account" type="p3ppee:purpose-role-value"/>
    <element name="arts" type="p3ppee:purpose-role-value"/>
    <element name="browsing" type="p3ppee:purpose-role-value"/>
    <element name="charity" type="p3ppee:purpose-role-value"/>
    <element name="communicate" type="p3ppee:purpose-role-value"/>
    <element name="custom" type="p3ppee:purpose-role-value"/>
    <element name="delivery" type="p3ppee:purpose-role-value"/>
    <element name="downloads" type="p3ppee:purpose-role-value"/>

    <element name="education" type="p3ppee:purpose-role-value"/>
    <element name="feedback" type="p3ppee:purpose-role-value"/>
    <element name="finmgt" type="p3ppee:purpose-role-value"/>
    <element name="gambling" type="p3ppee:purpose-role-value"/>
    <element name="gaming" type="p3ppee:purpose-role-value"/>
    <element name="government" type="p3ppee:purpose-role-value"/>
    <element name="health" type="p3ppee:purpose-role-value"/>
    <element name="login" type="p3ppee:purpose-role-value"/>
    <element name="marketing" type="p3ppee:purpose-role-value"/>

    <element name="news" type="p3ppee:purpose-role-value"/>
    <element name="payment" type="p3ppee:purpose-role-value"/>
```

```
      <element name="sales" type="p3ppee:purpose-role-value"/>
      <element name="search" type="p3ppee:purpose-role-value"/>
      <element name="state" type="p3ppee:purpose-role-value"/>
      <element name="surveys" type="p3ppee:purpose-role-value"/>
    </choice>
  </sequence>
 </complexType>

</element>

<!-- ********** Each purpose specified should contain at least one role********** -->
<complexType name="purpose-role-value">
        <sequence>
                <element ref="p3ppee:role" minOccurs="1" maxOccurs="unbounded"/>
        </sequence>
        <attribute name='required' use='optional' type='p3p:required-value'/>
</complexType>

<!-- ********** Employee Role ********** -->
<element name="role">
        <simpleType>
                <restriction base="string">
                        <enumeration value="MarketingAnalyst"/>
            <enumeration value="MarketingAdvisor"/>
            <enumeration value="EmailFlyerServer"/>
            <enumeration value="FlyerDistributor"/>
                        <enumeration value="WebServer"/>
                        <!-- ********** Customer Service Representative********** -->
                        <enumeration value="CSR"/>
                        <enumeration value="deliveryman"/>
                        <!-- ********** Web Customization Controller********** -->
                        <enumeration value="WCC"/>
                </restriction>
        </simpleType>
</element>

</schema>
```

Appendix 2: The Privacy Policies of Automart Defined in P3P 1.1 with P3PPEE.

1 p3p.xml

```
<META xmlns="http://www.w3.org/2002/01/P3Pv1">
 <POLICY-REFERENCES>
  <EXPIRY date="31 May 2011 05:00:00 GMT"/>

   <POLICY-REF about="/P3P/other_pages.xml#OtherPages">
    <INCLUDE>/*</INCLUDE>
    <EXCLUDE>/register.jsp</EXCLUDE>
    <EXCLUDE>/index.jsp</EXCLUDE>
    <EXCLUDE>/login.jsp</EXCLUDE>
    <EXCLUDE>/product.jsp</EXCLUDE>
    <EXCLUDE>/shopping_cart.jsp</EXCLUDE>
    <EXCLUDE>/order.jsp</EXCLUDE>
    <EXCLUDE>/historical_order_details.jsp</EXCLUDE>
   </POLICY-REF>

   <POLICY-REF about="/P3P/register_page.xml#RegisterPage">
    <INCLUDE>/register.jsp</INCLUDE>
   </POLICY-REF>

   <POLICY-REF about="/P3P/index_page.xml#IndexPage">
    <INCLUDE>/index.jsp</INCLUDE>
   </POLICY-REF>

   <POLICY-REF about="/P3P/shopping_cart_page.xml#ShoppingCartPage">
    <INCLUDE>/shopping_cart.jsp</INCLUDE>
   </POLICY-REF>

   <POLICY-REF about="/P3P/login_page.xml#LoginPage">
    <INCLUDE>/login.jsp</INCLUDE>
   </POLICY-REF>

   <POLICY-REF about="/P3P/product_page.xml#ProductPage">
    <INCLUDE>/product.jsp</INCLUDE>
   </POLICY-REF>

   <POLICY-REF about="/P3P/order_page.xml#OrderPage">
    <INCLUDE>/order.jsp</INCLUDE>
   </POLICY-REF>

   <POLICY-REF about="/P3P/historical_order_page.xml#HistoricalOrderPage">
    <INCLUDE>/historical_order_details.jsp</INCLUDE>
   </POLICY-REF>
 </POLICY-REFERENCES>
</META>
```

2 register_page.xml

```
<POLICIES xmlns="http://www.w3.org/2002/01/P3Pv1" xmlns:p3p11="http://www.w3.org/2006/01/P3Pv11"
xmlns:p3ppcee="http://www.example.org/P3PPCEE">
  <EXPIRY date="31 May 2011 05:00:00 GMT"/>
    <POLICY discuri="www.automart.com/privacy_policy.jsp" name="RegisterPage"
opturi="www.automart.com/register.jsp">
          <EXTENSION>
                  <p3p11:STAEMENT-GROUP-DEF id="account" consent = "mixed" short-description="Account
Registration Necessaries"/>
                  <p3p11:STAEMENT-GROUP-DEF id="profile" consent = "mixed" short-description="Account
Registration optionals"/>
                  <p3p11:STAEMENT-GROUP-DEF id="address" consent = "mixed" short-description="Account
Registration optionals"/>
                  <p3p11:STAEMENT-GROUP-DEF id="payment" consent = "mixed" short-description="Account
Registration optionals"/>
          </EXTENSION>
      <ENTITY>
        <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
          <DATA ref="#business.name">Automart</DATA>
          <DATA ref="#business.contact-info.postal.name.given">Dianshu</DATA>
          <DATA ref="#business.contact-info.postal.name.family">Hu</DATA>
          <DATA ref="#business.contact-info.postal.street">1680 Bank Street</DATA>
          <DATA ref="#business.contact-info.postal.city">Ottawa</DATA>
          <DATA ref="#business.contact-info.postal.stateprov">ON </DATA>
          <DATA ref="#business.contact-info.postal.postalcode">K3C6A8 </DATA>
          <DATA ref="#business.contact-info.postal.country">Canada</DATA>
          <DATA ref="#business.contact-info.online.email">hudianshu@automart.com</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.intcode">1</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.loccode">613</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.number">8888888</DATA>
        </DATA-GROUP>
        <EXTENSION>
          <p3p11:DATA-GROUP>
            <p3p11:datatype>
              <business>
                <orgname>Automart</orgname>
                <contact-info>
                  <postal>
                    <name>
                      <given>Dianshu</given>
                      <family>Hu</family>
                    </name>
                    <street>1680 Bank Street</street>
                    <city>Ottawa</city>
                    <stateprov>ON </stateprov>
                    <postalcode>K3C6A8 </postalcode>
                    <country>Canada</country>
                  </postal>
                  <online>
                    <email>hudianshu@automart.com</email>
                  </online>
                  <telecom>
                    <telephone>
```

```
                    <intcode>1</intcode>
                    <loccode>613</loccode>
                    <number>8888888</number>
                  </telephone>
                </telecom>
              </contact-info>
            </business>
          </p3p11:datatype>
        </p3p11:DATA-GROUP>
      </EXTENSION>
    </ENTITY>
    <ACCESS>
      <all/>
    </ACCESS>
    <DISPUTES-GROUP>
      <DISPUTES resolution-type="independent" service="http://www.truste.com/privacy_seals_and_services/" short-
description="TRUSTe" verification="">
        <IMG alt="TRUSTe's logo" src="http://www.truste.com/images/logo-truste.gif"/>
        <REMEDIES>
          <correct/>
        </REMEDIES>
      </DISPUTES>
      <DISPUTES resolution-type="service" service="www.automart.com/contact_us.jsp" short-description="Customer
Service" verification="">
        <REMEDIES>
          <correct/>
        </REMEDIES>
      </DISPUTES>
    </DISPUTES-GROUP>
    <STATEMENT>
      <EXTENSION>
            <p3p11:STATEMENT-GROUP id="account" />
      </EXTENSION>
      <CONSEQUENCE>We collect essential information to create an account for you and process your order. You need to
provide a username a password and your email address. </CONSEQUENCE>

      <PURPOSE>
        <EXTENSION>
          <PPURPOSE xmlns="http://www.w3.org/2004/02/P3Pv11">
            <account/>
          </PPURPOSE>
        </EXTENSION>
        <current/>
        <EXTENSION>
            <p3ppcee:PURPOSE-ROLES>
                <p3ppcee:account>
                    <p3ppcee:role>WebServer</p3ppcee:role>
                </p3ppcee:account>
            <p3ppcee:communicate>
                    <p3ppcee:role>WebServer</p3ppcee:role>
                <p3ppcee:role>CSR</p3ppcee:role>
                    </p3ppcee:communicate>
            </p3ppcee:PURPOSE-ROLES>
        </EXTENSION>
```

```
    </PURPOSE>
    <RECIPIENT>
      <ours>
              <recipient-description>www.automart.com</recipient-description>
      </ours>
    </RECIPIENT>
    <RETENTION>
       <EXTENSION xmlns:xf="http://www.w3.org/TR/2002/WD-xquery-operators-20020816/#" >
         <xf:dt-yearMonthDuration>
           P2Y
         </xf:dt-yearMonthDuration>
       </EXTENSION>
      <stated-purpose/>
    </RETENTION>
    <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
      <EXTENSION>
      <p3p11:data-group>
        <p3ppcee:datatype>
          <p3ppcee:dynamic>
           <p3ppcee:clickstream>
             <p3ppcee:timestamp/>
           </p3ppcee:clickstream>
          </p3ppcee:dynamic>
          <p3ppcee:user>
           <p3ppcee:login>
               <p3ppcee:id/>
               <p3ppcee:password/>
           </p3ppcee:login>
           <p3ppcee:home-info>
             <p3ppcee:online>
               <p3ppcee:email/>
             </p3ppcee:online>
           </p3ppcee:home-info>
          </p3ppcee:user>
        </p3ppcee:datatype>
      </p3p11:data-group>
    </EXTENSION>
      <DATA ref="#user.login.id"/>
      <DATA ref="#user.login.password"/>
      <DATA ref="#user.home-info.online.email"/>
      <DATA ref="#dynamic.clickstream.timestamp"/>
    </DATA-GROUP>
  </STATEMENT>

  <STATEMENT>
    <EXTENSION>
            <p3p11:STATEMENT-GROUP id="account" />
    </EXTENSION>
    <CONSEQUENCE>We collect essential information to create an account for you and process your order. You can
choose to receive flyers by email</CONSEQUENCE>

    <PURPOSE>
      <contact required="opt-out"/>
      <EXTENSION>
```

```
                <p3ppcee:PURPOSE-ROLES>
                        <p3ppcee:marketing required="opt-out">
                                <p3ppcee:role>MarketingAnalyst</p3ppcee:role>
                    <p3ppcee:role>MarketingAdvisor</p3ppcee:role>
                    <p3ppcee:role>EmailFlyerServer</p3ppcee:role>
                        </p3ppcee:marketing>
                </p3ppcee:PURPOSE-ROLES>
      </EXTENSION>
    </PURPOSE>
    <RECIPIENT>
      <ours>
                <recipient-description>www.automart.com</recipient-description>
      </ours>
      <same required="opt-out">
                <recipient-description>www.automodel.com</recipient-description>
      </same>
    </RECIPIENT>
    <RETENTION>
       <EXTENSION xmlns:xf="http://www.w3.org/TR/2002/WD-xquery-operators-20020816/#" >
         <xf:dt-yearMonthDuration>
           P2Y
         </xf:dt-yearMonthDuration>
       </EXTENSION>
      <stated-purpose/>
    </RETENTION>
    <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
      <EXTENSION>
      <p3p11:data-group>
        <p3ppcee:datatype>
          <p3ppcee:user>
            <p3ppcee:home-info>
              <p3ppcee:online>
                <p3ppcee:email/>
              </p3ppcee:online>
            </p3ppcee:home-info>
          </p3ppcee:user>
        </p3ppcee:datatype>
      </p3p11:data-group>
    </EXTENSION>
      <DATA ref="#user.home-info.online.email"/>
    </DATA-GROUP>

  </STATEMENT>

  <STATEMENT>
      <EXTENSION>
                <p3p11:STATEMENT-GROUP id="profile" />
      </EXTENSION>
      <CONSEQUENCE>You can choose to save additional demographic information with your account thus we can serve
you better. With your consent, we may distribute the data to the specified business partners.</CONSEQUENCE>

    <PURPOSE>
      <EXTENSION>
        <PPURPOSE xmlns="http://www.w3.org/2004/02/P3Pv11">
```

```
      <account/>
      <feedback/>
    </PPURPOSE>
  </EXTENSION>
  <current/>
  <contact required="opt-out"/>
  <EXTENSION>
          <p3ppcee:PURPOSE-ROLES>
                  <p3ppcee:account>
                          <p3ppcee:role>WebServer</p3ppcee:role>
                  </p3ppcee:account>
                  <p3ppcee:feedback>
                          <p3ppcee:role>CSR</p3ppcee:role>
                  </p3ppcee:feedback>
                  <p3ppcee:marketing required="opt-out">
                          <p3ppcee:role>MarketingAnalyst</p3ppcee:role>
              <p3ppcee:role>MarketingAdvisor</p3ppcee:role>
                  </p3ppcee:marketing>
          </p3ppcee:PURPOSE-ROLES>
  </EXTENSION>
</PURPOSE>
<RECIPIENT>
  <ours>
          <recipient-description>www.automart.com</recipient-description>
  </ours>
  <same required="opt-out">
          <recipient-description>www.automodel.com</recipient-description>
  </same>
</RECIPIENT>
<RETENTION>
   <EXTENSION xmlns:xf="http://www.w3.org/TR/2002/WD-xquery-operators-20020816/#" >
      <xf:dt-yearMonthDuration>
        P2Y
      </xf:dt-yearMonthDuration>
    </EXTENSION>
  <stated-purpose/>
</RETENTION>
<DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
   <EXTENSION>
  <p3p11:data-group>
    <p3ppcee:datatype>
      <p3ppcee:user>
        <p3ppcee:name>
          <p3ppcee:given/>
          <p3ppcee:middle/>
          <p3ppcee:family/>
        </p3ppcee:name>
        <p3ppcee:bdate/>
        <p3ppcee:gender/>
      </p3ppcee:user>
    </p3ppcee:datatype>
  </p3p11:data-group>
</EXTENSION>
  <DATA ref="#user.name.given" optional="yes"/>
```

```
    <DATA ref="#user.name.middle" optional="yes"/>
    <DATA ref="#user.name.family" optional="yes"/>
    <DATA ref="#user.bdate" optional="yes"/>
    <DATA ref="#user.gender" optional="yes"/>
  </DATA-GROUP>
</STATEMENT>

<STATEMENT>
  <EXTENSION>
        <p3p11:STATEMENT-GROUP id="address" />
  </EXTENSION>
  <CONSEQUENCE>You can choose to save additional postal address information with your account thus we can serve
you better and faster. With your consent, we may distribute the data to the specified business partners.</CONSEQUENCE>

  <PURPOSE>
    <EXTENSION>
      <PPURPOSE xmlns="http://www.w3.org/2004/02/P3Pv11">
        <account/>
        <payment/>
        <feedback/>
        <delivery/>
      </PPURPOSE>
    </EXTENSION>
    <contact required="opt-out"/>
    <current/>
    <EXTENSION>
        <p3ppcee:PURPOSE-ROLES>
            <p3ppcee:account>
                <p3ppcee:role>WebServer</p3ppcee:role>
            </p3ppcee:account>
            <p3ppcee:payment>
                <p3ppcee:role>WebServer</p3ppcee:role>
            </p3ppcee:payment>
            <p3ppcee:delivery>
                <p3ppcee:role>deliveryman</p3ppcee:role>
            </p3ppcee:delivery>
            <p3ppcee:communicate>
                <p3ppcee:role>CSR</p3ppcee:role>
            </p3ppcee:communicate>
            <p3ppcee:marketing required="opt-out">
                <p3ppcee:role>MarketingAnalyst</p3ppcee:role>
          <p3ppcee:role>MarketingAdvisor</p3ppcee:role>
          <p3ppcee:role>FlyerDistributor</p3ppcee:role>
            </p3ppcee:marketing>
        </p3ppcee:PURPOSE-ROLES>
    </EXTENSION>
  </PURPOSE>
  <RECIPIENT>
    <ours>
        <recipient-description>www.automart.com</recipient-description>
    </ours>
    <same required="opt-out">
        <recipient-description>www.automodel.com</recipient-description>
    </same>
```

```
    </RECIPIENT>
    <RETENTION>
      <EXTENSION xmlns:xf="http://www.w3.org/TR/2002/WD-xquery-operators-20020816/#" >
          <xf:dt-yearMonthDuration>
            P2Y
          </xf:dt-yearMonthDuration>
        </EXTENSION>
      <stated-purpose/>
    </RETENTION>
    <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
      <EXTENSION>
      <p3p11:data-group>
        <p3ppcee:datatype>
          <p3ppcee:user>
            <p3ppcee:home-info>
              <p3ppcee:postal>
                <p3ppcee:street/>
                <p3ppcee:city/>
                <p3ppcee:stateprov/>
                <p3ppcee:postalcode/>
                <p3ppcee:country/>
              </p3ppcee:postal>
            </p3ppcee:home-info>
          </p3ppcee:user>
        </p3ppcee:datatype>
      </p3p11:data-group>
    </EXTENSION>
      <DATA ref="#user.home-info.postal.street" optional="yes"/>
      <DATA ref="#user.home-info.postal.city" optional="yes"/>
      <DATA ref="#user.home-info.postal.stateprov" optional="yes"/>
      <DATA ref="#user.home-info.postal.postalcode" optional="yes"/>
      <DATA ref="#user.home-info.postal.country" optional="yes"/>
      </DATA-GROUP>
  </STATEMENT>

  <STATEMENT>
      <EXTENSION>
              <p3p11:STATEMENT-GROUP id="payment" />
      </EXTENSION>
      <CONSEQUENCE>You can choose to save additional payment information with your account thus we can serve you
faster. With your consent, we may distribute the data to the specified business partners.</CONSEQUENCE>

    <PURPOSE>
      <EXTENSION>
        <PPURPOSE xmlns="http://www.w3.org/2004/02/P3Pv11">
          <sales/>
          <account/>
          <payment/>
          <marketing/>
        </PPURPOSE>
      </EXTENSION>
      <current/>
      <EXTENSION>
              <p3ppcee:PURPOSE-ROLES>
```

```
                    <p3ppcee:account>
                        <p3ppcee:role>WebServer</p3ppcee:role>
                    </p3ppcee:account>
                    <p3ppcee:sales>
                        <p3ppcee:role>WebServer</p3ppcee:role>
                    </p3ppcee:sales>
                    <p3ppcee:payment>
                        <p3ppcee:role>WebServer</p3ppcee:role>
                    </p3ppcee:payment>
                    <p3ppcee:marketing required="opt-out">
                        <p3ppcee:role>MarketingAnalyst</p3ppcee:role>
                <p3ppcee:role>MarketingAdvisor</p3ppcee:role>
                    </p3ppcee:marketing>
            </p3ppcee:PURPOSE-ROLES>
  </EXTENSION>
 </PURPOSE>
 <RECIPIENT>
  <ours>
            <recipient-description>www.automart.com</recipient-description>
  </ours>
  <same required="opt-out">
            <recipient-description>www.automodel.com</recipient-description>
  </same>
 </RECIPIENT>
 <RETENTION>
   <EXTENSION xmlns:xf="http://www.w3.org/TR/2002/WD-xquery-operators-20020816/#" >
      <xf:dt-yearMonthDuration>
        P2Y
      </xf:dt-yearMonthDuration>
    </EXTENSION>
  <stated-purpose/>
 </RETENTION>
 <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
   <EXTENSION>
  <p3p11:data-group>
    <p3ppcee:datatype>
      <p3ppcee:user>
            <p3ppcee:payment>
                    <p3ppcee:creditcard>
                        <p3ppcee:cctype/>
                        <p3ppcee:ccnum/>
                        <p3ppcee:holdername/>
                        <p3ppcee:expiry/>
                        <p3ppcee:securitycode/>
                    </p3ppcee:creditcard>
            </p3ppcee:payment>
      </p3ppcee:user>
    </p3ppcee:datatype>
  </p3p11:data-group>
</EXTENSION>
            <DATA ref="#user.payment.creditcard.cctype" optional="yes"/>
            <DATA ref="#user.payment.creditcard.ccnum" optional="yes"/>
            <DATA ref="#user.payment.creditcard.holdername" optional="yes"/>
            <DATA ref="#user.payment.creditcard.expiry" optional="yes"/>
```

```
            <DATA ref="#user.payment.creditcard.securitycode" optional="yes"/>
        </DATA-GROUP>
      </STATEMENT>
    </POLICY>
  </POLICIES>
```

3 index_page.xml

```
<POLICIES xmlns="http://www.w3.org/2002/01/P3Pv1" xmlns:p3p11="http://www.w3.org/2006/01/P3Pv11">
    <EXPIRY date="31 May 2011 05:00:00 GMT"/>
    <POLICY discuri="www.automart.com/privacy_policy.jsp" name="IndexPage">
      <ENTITY>
        <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
          <DATA ref="#business.name">Automart</DATA>
          <DATA ref="#business.contact-info.postal.name.given">Dianshu</DATA>
          <DATA ref="#business.contact-info.postal.name.family">Hu</DATA>
          <DATA ref="#business.contact-info.postal.street">1680 Bank Street</DATA>
          <DATA ref="#business.contact-info.postal.city">Ottawa</DATA>
          <DATA ref="#business.contact-info.postal.stateprov">ON </DATA>
          <DATA ref="#business.contact-info.postal.postalcode">K3C6A8 </DATA>
          <DATA ref="#business.contact-info.postal.country">Canada</DATA>
          <DATA ref="#business.contact-info.online.email">hudianshu@automart.com</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.intcode">1</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.loccode">613</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.number">8888888</DATA>
        </DATA-GROUP>
        <EXTENSION>
          <p3p11:DATA-GROUP>
            <p3p11:datatype>
              <business>
                <orgname>Automart</orgname>
                <contact-info>
                  <postal>
                    <name>
                      <given>Dianshu</given>
                      <family>Hu</family>
                    </name>
                    <street>1680 Bank Street</street>
                    <city>Ottawa</city>
                    <stateprov>ON </stateprov>
                    <postalcode>K3C6A8 </postalcode>
                    <country>Canada</country>
                  </postal>
                  <online>
                    <email>hudianshu@automart.com</email>
                  </online>
                  <telecom>
                    <telephone>
                      <intcode>1</intcode>
                      <loccode>613</loccode>
                      <number>8888888</number>
                    </telephone>
                  </telecom>
                </contact-info>
              </business>
            </p3p11:datatype>
          </p3p11:DATA-GROUP>
        </EXTENSION>
      </ENTITY>
      <ACCESS>
        <nonident/>
```

```
    </ACCESS>
    <DISPUTES-GROUP>
      <DISPUTES resolution-type="independent" service="http://www.truste.com/privacy_seals_and_services/" short-
description="TRUSTe" verification="">
        <IMG alt="TRUSTe's logo" src="http://www.truste.com/images/logo-truste.gif"/>
        <REMEDIES>
          <correct/>
        </REMEDIES>
      </DISPUTES>
      <DISPUTES resolution-type="service" service="www.automart.com/contact_us.jsp" short-description="Customer
Service" verification="">
        <REMEDIES>
          <correct/>
        </REMEDIES>
      </DISPUTES>
    </DISPUTES-GROUP>
    <STATEMENT>
      <CONSEQUENCE>We record your user agent information stored in http request header improve our web site. We also
collects your ip and timestamp of http request for security purpose. </CONSEQUENCE>
      <PURPOSE>
        <develop required="always"/>
        <admin required="always"/>
      </PURPOSE>
      <RECIPIENT>
        <ours/>
      </RECIPIENT>
      <RETENTION>
        <stated-purpose/>
      </RETENTION>
      <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
        <EXTENSION>
        <p3p11:data-group>
          <p3p11:datatype>
            <p3p11:dynamic>
              <p3p11:clickstream>
                 <p3p11:timestamp/>
                 <p3p11:clientip/>
              </p3p11:clickstream>
              <p3p11:http>
                <p3p11:useragent/>
              </p3p11:http>
            </p3p11:dynamic>
          </p3p11:datatype>
        </p3p11:data-group>
        </EXTENSION>
        <DATA ref="#dynamic.clickstream.clientip"/>
        <DATA ref="#dynamic.clickstream.timestamp"/>
        <DATA ref="#dynamic.http.useragent"/>

      </DATA-GROUP>
    </STATEMENT>
    <STATEMENT>
      <CONSEQUENCE>You clickstream is collected on this page to show the stock cars of the category selected by youself.
</CONSEQUENCE>
      <PURPOSE>
        <EXTENSION>
          <PPURPOSE xmlns="http://www.w3.org/2004/02/P3Pv11">
            <browsing/>
          </PPURPOSE>
        </EXTENSION>
        <current/>
      </PURPOSE>
```

```
      <RECIPIENT>
        <ours/>
      </RECIPIENT>
      <RETENTION>
        <no-retention/>
      </RETENTION>
      <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
        <EXTENSION>
        <p3p11:data-group>
          <p3p11:datatype>
            <p3p11:dynamic>
              <p3p11:clickstream>
                 <p3p11:uri/>
              </p3p11:clickstream>
            </p3p11:dynamic>
          </p3p11:datatype>
        </p3p11:data-group>
      </EXTENSION>
        <DATA ref="#dynamic.clickstream.uri"/>
      </DATA-GROUP>
    </STATEMENT>
  </POLICY>
</POLICIES>
```

4 login_page.xml

```
<POLICIES xmlns="http://www.w3.org/2002/01/P3Pv1" xmlns:p3p11="http://www.w3.org/2006/01/P3Pv11">
  <EXPIRY date="31 May 2011 05:00:00 GMT"/>
    <POLICY discuri="www.automart.com/privacy_policy.jsp" name="LoginPage" opturi="www.automart.com/login.jsp">
      <ENTITY>
        <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
          <DATA ref="#business.name">Automart</DATA>
          <DATA ref="#business.contact-info.postal.name.given">Dianshu</DATA>
          <DATA ref="#business.contact-info.postal.name.family">Hu</DATA>
          <DATA ref="#business.contact-info.postal.street">1680 Bank Street</DATA>
          <DATA ref="#business.contact-info.postal.city">Ottawa</DATA>
          <DATA ref="#business.contact-info.postal.stateprov">ON </DATA>
          <DATA ref="#business.contact-info.postal.postalcode">K3C6A8 </DATA>
          <DATA ref="#business.contact-info.postal.country">Canada</DATA>
          <DATA ref="#business.contact-info.online.email">hudianshu@automart.com</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.intcode">1</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.loccode">613</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.number">8888888</DATA>
        </DATA-GROUP>
        <EXTENSION>
          <DATA-GROUP xmlns="http://www.w3.org/2004/02/P3Pv11">
            <datatype xmlns="http://www.w3.org/2002/01/BDSP3Pv1.1">
              <business>
                <orgname>Automart</orgname>
                <contact-info>
                  <postal>
                    <name>
                      <given>Dianshu</given>
                      <family>Hu</family>
                    </name>
                    <street>1680 Bank Street</street>
                    <city>Ottawa</city>
                    <stateprov>ON </stateprov>
                    <postalcode>K3C6A8 </postalcode>
                    <country>Canada</country>
```

```
                </postal>
                <online>
                  <email>hudianshu@automart.com</email>
                </online>
                <telecom>
                  <telephone>
                    <intcode>1</intcode>
                    <loccode>613</loccode>
                    <number>8888888</number>
                  </telephone>
                </telecom>
              </contact-info>
            </business>
          </datatype>
        </DATA-GROUP>
      </EXTENSION>
    </ENTITY>
    <ACCESS>
      <all/>
    </ACCESS>
    <DISPUTES-GROUP>
      <DISPUTES resolution-type="independent" service="http://www.truste.com/privacy_seals_and_services/" short-
description="TRUSTe" verification="">
        <IMG alt="TRUSTe's logo" src="http://www.truste.com/images/logo-truste.gif"/>
        <REMEDIES>
          <correct/>
        </REMEDIES>
      </DISPUTES>
      <DISPUTES resolution-type="service" service="www.automart.com/contact_us.jsp" short-description="Customer
Service" verification="">
        <REMEDIES>
          <correct/>
        </REMEDIES>
      </DISPUTES>
    </DISPUTES-GROUP>
    <STATEMENT>
      <CONSEQUENCE>login id and password are collected on this page in order to authenticate
customer</CONSEQUENCE>
      <PURPOSE>
        <EXTENSION>
          <PPURPOSE xmlns="http://www.w3.org/2004/02/P3Pv11">
            <login required="opt-in"/>
          </PPURPOSE>
        </EXTENSION>
        <current/>
      </PURPOSE>
      <RECIPIENT>
        <ours/>
      </RECIPIENT>
      <RETENTION>
        <no-retention/>
      </RETENTION>
      <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
        <EXTENSION>
        <p3p11:data-group>
          <p3p11:datatype>
            <p3p11:user>
              <p3p11:login/>
            </p3p11:user>
          </p3p11:datatype>
        </p3p11:data-group>
      </EXTENSION>
```

```
        <DATA ref="#user.login"/>
      </DATA-GROUP>
    </STATEMENT>
  </POLICY>
</POLICIES>
```

5 product_page.xml

```
<POLICIES xmlns="http://www.w3.org/2002/01/P3Pv1" xmlns:p3p11="http://www.w3.org/2006/01/P3Pv11">
  <EXPIRY date="31 May 2011 05:00:00 GMT"/>
    <POLICY discuri="/privacy_policy.jsp" name="ProductPage">
      <ENTITY>
        <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
          <DATA ref="#business.name">Automart</DATA>
          <DATA ref="#business.contact-info.postal.name.given">Dianshu</DATA>
          <DATA ref="#business.contact-info.postal.name.family">Hu</DATA>
          <DATA ref="#business.contact-info.postal.street">1680 Bank Street</DATA>
          <DATA ref="#business.contact-info.postal.city">Ottawa</DATA>
          <DATA ref="#business.contact-info.postal.stateprov">ON </DATA>
          <DATA ref="#business.contact-info.postal.postalcode">K3C6A8 </DATA>
          <DATA ref="#business.contact-info.postal.country">Canada</DATA>
          <DATA ref="#business.contact-info.online.email">hudianshu@automart.com</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.intcode">1</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.loccode">613</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.number">8888888</DATA>
        </DATA-GROUP>
        <EXTENSION>
          <DATA-GROUP xmlns="http://www.w3.org/2004/02/P3Pv11">
            <datatype xmlns="http://www.w3.org/2002/01/BDSP3Pv1.1">
              <business>
                <orgname>Automart</orgname>
                <contact-info>
                  <postal>
                    <name>
                      <given>Dianshu</given>
                      <family>Hu</family>
                    </name>
                    <street>1680 Bank Street</street>
                    <city>Ottawa</city>
                    <stateprov>ON </stateprov>
                    <postalcode>K3C6A8 </postalcode>
                    <country>Canada</country>
                  </postal>
                  <online>
                    <email>hudianshu@automart.com</email>
                  </online>
                  <telecom>
                    <telephone>
                      <intcode>1</intcode>
                      <loccode>613</loccode>
                      <number>8888888</number>
                    </telephone>
                  </telecom>
                </contact-info>
              </business>
            </datatype>
          </DATA-GROUP>
        </EXTENSION>
      </ENTITY>
      <ACCESS>
```

```
        <nonident/>
      </ACCESS>
      <DISPUTES-GROUP>
        <DISPUTES resolution-type="independent" service="http://www.truste.com/privacy_seals_and_services/" short-
description="TRUSTe" verification="">
          <IMG alt="TRUSTe's logo" src="http://www.truste.com/images/logo-truste.gif"/>
          <REMEDIES>
            <correct/>
          </REMEDIES>
        </DISPUTES>
        <DISPUTES resolution-type="service" service="www.automart.com/contact_us.jsp" short-description="Customer
Service" verification="">
          <REMEDIES>
            <correct/>
          </REMEDIES>
        </DISPUTES>
      </DISPUTES-GROUP>
      <STATEMENT>
        <CONSEQUENCE>On product page, only the query-string portion of uri to this page is collected for retrieving the
vehicle detail and one-time tailoring</CONSEQUENCE>
        <PURPOSE>
          <EXTENSION>
            <PPURPOSE xmlns="http://www.w3.org/2004/02/P3Pv11">
              <browsing/>
            </PPURPOSE>
          </EXTENSION>
          <tailoring/>
          <current/>
        </PURPOSE>
        <RECIPIENT>
          <ours/>
        </RECIPIENT>
        <RETENTION>
          <no-retention/>
        </RETENTION>
        <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
          <EXTENSION>
          <p3p11:data-group>
            <p3p11:datatype>
              <p3p11:dynamic>
                <p3p11:clickstream>
                  <p3p11:uri>
                    <p3p11:querystring/>
                  </p3p11:uri>
                </p3p11:clickstream>
              </p3p11:dynamic>
            </p3p11:datatype>
          </p3p11:data-group>
        </EXTENSION>
          <DATA ref="#dynamic.clickstream.uri.querystring"/>
        </DATA-GROUP>
      </STATEMENT>
    </POLICY>
  </POLICIES>
```

6 order_page.xml

```
<POLICIES xmlns="http://www.w3.org/2002/01/P3Pv1" xmlns:p3p11="http://www.w3.org/2006/01/P3Pv11">
  <EXPIRY date="31 May 2011 05:00:00 GMT"/>
    <POLICY discuri="www.automart.com/privacy_policy.jsp" name="OrderPage">
```

```
<ENTITY>
  <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
    <DATA ref="#business.name">Automart</DATA>
    <DATA ref="#business.contact-info.postal.name.given">Dianshu</DATA>
    <DATA ref="#business.contact-info.postal.name.family">Hu</DATA>
    <DATA ref="#business.contact-info.postal.street">1680 Bank Street</DATA>
    <DATA ref="#business.contact-info.postal.city">Ottawa</DATA>
    <DATA ref="#business.contact-info.postal.stateprov">ON </DATA>
    <DATA ref="#business.contact-info.postal.postalcode">K3C6A8 </DATA>
    <DATA ref="#business.contact-info.postal.country">Canada</DATA>
    <DATA ref="#business.contact-info.online.email">hudianshu@automart.com</DATA>
    <DATA ref="#business.contact-info.telecom.telephone.intcode">1</DATA>
    <DATA ref="#business.contact-info.telecom.telephone.loccode">613</DATA>
    <DATA ref="#business.contact-info.telecom.telephone.number">8888888</DATA>
  </DATA-GROUP>
  <EXTENSION>
    <DATA-GROUP xmlns="http://www.w3.org/2004/02/P3Pv11">
      <datatype xmlns="http://www.w3.org/2002/01/BDSP3Pv1.1">
        <business>
          <orgname>Automart</orgname>
          <contact-info>
            <postal>
              <name>
                <given>Dianshu</given>
                <family>Hu</family>
              </name>
              <street>1680 Bank Street</street>
              <city>Ottawa</city>
              <stateprov>ON </stateprov>
              <postalcode>K3C6A8 </postalcode>
              <country>Canada</country>
            </postal>
            <online>
              <email>hudianshu@automart.com</email>
            </online>
            <telecom>
              <telephone>
                <intcode>1</intcode>
                <loccode>613</loccode>
                <number>8888888</number>
              </telephone>
            </telecom>
          </contact-info>
        </business>
      </datatype>
    </DATA-GROUP>
  </EXTENSION>
</ENTITY>
<ACCESS>
  <nonident/>
</ACCESS>
<DISPUTES-GROUP>
  <DISPUTES resolution-type="independent" service="http://www.truste.com/privacy_seals_and_services/" short-
description="TRUSTe" verification="">
    <IMG alt="TRUSTe's logo" src="http://www.truste.com/images/logo-truste.gif"/>
    <REMEDIES>
      <correct/>
    </REMEDIES>
  </DISPUTES>
  <DISPUTES resolution-type="service" service="www.automart.com/contact_us.jsp" short-description="Customer
Service" verification="">
    <REMEDIES>
```

```
            <correct/>
          </REMEDIES>
        </DISPUTES>
      </DISPUTES-GROUP>
      <STATEMENT>
        <CONSEQUENCE>The order page collects timestamps</CONSEQUENCE>
        <PURPOSE>
          <EXTENSION>
            <PPURPOSE xmlns="http://www.w3.org/2004/02/P3Pv11">
              <sales/>
            </PPURPOSE>
          </EXTENSION>
          <current/>
        </PURPOSE>
        <RECIPIENT>
          <ours/>
        </RECIPIENT>
        <RETENTION>
          <stated-purpose/>
        </RETENTION>
        <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
          <EXTENSION>
          <p3p11:data-group>
            <p3p11:datatype>
              <p3p11:dynamic>
                <p3p11:clickstream>
                  <p3p11:timestamp/>
                </p3p11:clickstream>
              </p3p11:dynamic>
            </p3p11:datatype>
          </p3p11:data-group>
        </EXTENSION>
          <DATA ref="#dynamic.clickstream.timestamp"/>
        </DATA-GROUP>
      </STATEMENT>
    </POLICY>
  </POLICIES>
```

7 historical_order_page.xml

```
<POLICIES xmlns="http://www.w3.org/2002/01/P3Pv1" xmlns:p3p11="http://www.w3.org/2006/01/P3Pv11">
  <EXPIRY date="31 May 2011 05:00:00 GMT"/>
    <POLICY discuri="www.automart.com/privacy_policy.jsp" name="HistoricalOrderPage">
      <ENTITY>
        <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
          <DATA ref="#business.name">Automart</DATA>
          <DATA ref="#business.contact-info.postal.name.given">Dianshu</DATA>
          <DATA ref="#business.contact-info.postal.name.family">Hu</DATA>
          <DATA ref="#business.contact-info.postal.street">1680 Bank Street</DATA>
          <DATA ref="#business.contact-info.postal.city">Ottawa</DATA>
          <DATA ref="#business.contact-info.postal.stateprov">ON </DATA>
          <DATA ref="#business.contact-info.postal.postalcode">K3C6A8 </DATA>
          <DATA ref="#business.contact-info.postal.country">Canada</DATA>
          <DATA ref="#business.contact-info.online.email">hudianshu@automart.com</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.intcode">1</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.loccode">613</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.number">8888888</DATA>
        </DATA-GROUP>
        <EXTENSION>
          <DATA-GROUP xmlns="http://www.w3.org/2004/02/P3Pv11">
```

```
            <datatype xmlns="http://www.w3.org/2002/01/BDSP3Pv1.1">
              <business>
                <orgname>Automart</orgname>
                <contact-info>
                  <postal>
                    <name>
                      <given>Dianshu</given>
                      <family>Hu</family>
                    </name>
                    <street>1680 Bank Street</street>
                    <city>Ottawa</city>
                    <stateprov>ON </stateprov>
                    <postalcode>K3C6A8 </postalcode>
                    <country>Canada</country>
                  </postal>
                  <online>
                    <email>hudianshu@automart.com</email>
                  </online>
                  <telecom>
                    <telephone>
                      <intcode>1</intcode>
                      <loccode>613</loccode>
                      <number>8888888</number>
                    </telephone>
                  </telecom>
                </contact-info>
              </business>
            </datatype>
          </DATA-GROUP>
        </EXTENSION>
      </ENTITY>
      <ACCESS>
        <nonident/>
      </ACCESS>
      <DISPUTES-GROUP>
        <DISPUTES resolution-type="independent" service="http://www.truste.com/privacy_seals_and_services/" short-
description="TRUSTe" verification="">
          <IMG alt="TRUSTe's logo" src="http://www.truste.com/images/logo-truste.gif"/>
          <REMEDIES>
            <correct/>
          </REMEDIES>
        </DISPUTES>
        <DISPUTES resolution-type="service" service="www.automart.com/contact_us.jsp" short-description="Customer
Service" verification="">
          <REMEDIES>
            <correct/>
          </REMEDIES>
        </DISPUTES>
      </DISPUTES-GROUP>
      <STATEMENT>
        <CONSEQUENCE>On historical_order page, only the query string of the URI is collected for retrieving the details of
the selected historical order</CONSEQUENCE>
          <PURPOSE>
            <EXTENSION>
              <PPURPOSE xmlns="http://www.w3.org/2004/02/P3Pv11">
                <browsing/>
              </PPURPOSE>
            </EXTENSION>
            <current/>
          </PURPOSE>
          <RECIPIENT>
            <ours/>
```

```
      </RECIPIENT>
      <RETENTION>
        <no-retention/>
      </RETENTION>
      <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
        <EXTENSION>
        <p3p11:data-group>
          <p3p11:datatype>
            <p3p11:dynamic>
              <p3p11:clickstream>
                <p3p11:uri>
                  <p3p11:querystring/>
                </p3p11:uri>
              </p3p11:clickstream>
            </p3p11:dynamic>
          </p3p11:datatype>
        </p3p11:data-group>
      </EXTENSION>
        <DATA ref="#dynamic.clickstream.uri.querystring"/>
      </DATA-GROUP>
    </STATEMENT>
  </POLICY>
</POLICIES>
```

8 other_pages.xml

```
<POLICIES xmlns="http://www.w3.org/2002/01/P3Pv1" xmlns:p3p11="http://www.w3.org/2006/01/P3Pv11">
  <EXPIRY date="31 May 2011 05:00:00 GMT"/>
    <POLICY discuri="/privacy_policy.jsp" name="OtherPages">
      <ENTITY>
        <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
          <DATA ref="#business.name">Automart</DATA>
          <DATA ref="#business.contact-info.postal.name.given">Dianshu</DATA>
          <DATA ref="#business.contact-info.postal.name.family">Hu</DATA>
          <DATA ref="#business.contact-info.postal.street">1680 Bank Street</DATA>
          <DATA ref="#business.contact-info.postal.city">Ottawa</DATA>
          <DATA ref="#business.contact-info.postal.stateprov">ON </DATA>
          <DATA ref="#business.contact-info.postal.postalcode">K3C6A8 </DATA>
          <DATA ref="#business.contact-info.postal.country">Canada</DATA>
          <DATA ref="#business.contact-info.online.email">hudianshu@automart.com</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.intcode">1</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.loccode">613</DATA>
          <DATA ref="#business.contact-info.telecom.telephone.number">8888888</DATA>
        </DATA-GROUP>
        <EXTENSION>
          <DATA-GROUP xmlns="http://www.w3.org/2004/02/P3Pv11">
            <datatype xmlns="http://www.w3.org/2002/01/BDSP3Pv1.1">
              <business>
                <orgname>Automart</orgname>
                <contact-info>
                  <postal>
                    <name>
                      <given>Dianshu</given>
                      <family>Hu</family>
                    </name>
                    <street>1680 Bank Street</street>
                    <city>Ottawa</city>
                    <stateprov>ON </stateprov>
                    <postalcode>K3C6A8 </postalcode>
                    <country>Canada</country>
```

```
            </postal>
            <online>
              <email>hudianshu@automart.com</email>
            </online>
            <telecom>
              <telephone>
                <intcode>1</intcode>
                <loccode>613</loccode>
                <number>8888888</number>
              </telephone>
            </telecom>
          </contact-info>
        </business>
      </datatype>
    </DATA-GROUP>
  </EXTENSION>
</ENTITY>
<ACCESS>
  <nonident/>
</ACCESS>
<DISPUTES-GROUP>
  <DISPUTES resolution-type="independent" service="http://www.truste.com/privacy_seals_and_services/" short-
description="TRUSTe" verification="">
    <IMG alt="TRUSTe's logo" src="http://www.truste.com/images/logo-truste.gif"/>
    <REMEDIES>
      <correct/>
    </REMEDIES>
  </DISPUTES>
  <DISPUTES resolution-type="service" service="www.automart.com/contact_us.jsp" short-description="Customer
Service" verification="">
    <REMEDIES>
      <correct/>
    </REMEDIES>
  </DISPUTES>
</DISPUTES-GROUP>
<STATEMENT>
  <CONSEQUENCE>This policy cover the rest of the online store. It specifies that no identifiable data is
collected</CONSEQUENCE>
  <NON-IDENTIFIABLE/>
</STATEMENT>
</POLICY>
</POLICIES>
```

9 shopping_cart.xml

```
<POLICIES xmlns="http://www.w3.org/2002/01/P3Pv1" xmlns:p3p11="http://www.w3.org/2006/01/P3Pv11">
  <EXPIRY date="31 May 2011 05:00:00 GMT"/>
  <POLICY discuri="www.automart.com/privacy_policy.jsp" name="ShoppingCartPage">
    <ENTITY>
      <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
        <DATA ref="#business.name">Automart</DATA>
        <DATA ref="#business.contact-info.postal.name.given">Dianshu</DATA>
        <DATA ref="#business.contact-info.postal.name.family">Hu</DATA>
        <DATA ref="#business.contact-info.postal.street">1680 Bank Street</DATA>
        <DATA ref="#business.contact-info.postal.city">Ottawa</DATA>
        <DATA ref="#business.contact-info.postal.stateprov">ON </DATA>
        <DATA ref="#business.contact-info.postal.postalcode">K3C6A8 </DATA>
        <DATA ref="#business.contact-info.postal.country">Canada</DATA>
        <DATA ref="#business.contact-info.online.email">hudianshu@automart.com</DATA>
        <DATA ref="#business.contact-info.telecom.telephone.intcode">1</DATA>
        <DATA ref="#business.contact-info.telecom.telephone.loccode">613</DATA>
```

```
            <DATA ref="#business.contact-info.telecom.telephone.number">8888888</DATA>
          </DATA-GROUP>
          <EXTENSION>
            <DATA-GROUP xmlns="http://www.w3.org/2004/02/P3Pv11">
              <datatype xmlns="http://www.w3.org/2002/01/BDSP3Pv1.1">
                <business>
                  <orgname>Automart</orgname>
                  <contact-info>
                    <postal>
                      <name>
                        <given>Dianshu</given>
                        <family>Hu</family>
                      </name>
                      <street>1680 Bank Street</street>
                      <city>Ottawa</city>
                      <stateprov>ON </stateprov>
                      <postalcode>K3C6A8 </postalcode>
                      <country>Canada</country>
                    </postal>
                    <online>
                      <email>hudianshu@automart.com</email>
                    </online>
                    <telecom>
                      <telephone>
                        <intcode>1</intcode>
                        <loccode>613</loccode>
                        <number>8888888</number>
                      </telephone>
                    </telecom>
                  </contact-info>
                </business>
              </datatype>
            </DATA-GROUP>
          </EXTENSION>
        </ENTITY>
        <ACCESS>
          <nonident/>
        </ACCESS>
        <DISPUTES-GROUP>
          <DISPUTES resolution-type="independent" service="http://www.truste.com/privacy_seals_and_services/" short-
description="TRUSTe" verification="">
            <IMG alt="TRUSTe's logo" src="http://www.truste.com/images/logo-truste.gif"/>
            <REMEDIES>
              <correct/>
            </REMEDIES>
          </DISPUTES>
          <DISPUTES resolution-type="service" service="www.automart.com/contact_us.jsp" short-description="Customer
Service" verification="">
            <REMEDIES>
              <correct/>
            </REMEDIES>
          </DISPUTES>
        </DISPUTES-GROUP>
        <STATEMENT>
          <CONSEQUENCE>On product page, only the IDs of the products chosen by customer are collected to remove the
products from shopping cart</CONSEQUENCE>
          <PURPOSE>
            <EXTENSION>
              <PPURPOSE xmlns="http://www.w3.org/2004/02/P3Pv11">
                <sales/>
              </PPURPOSE>
            </EXTENSION>
```

```
      <current/>
    </PURPOSE>
    <RECIPIENT>
      <ours/>
    </RECIPIENT>
    <RETENTION>
      <no-retention/>
    </RETENTION>
    <DATA-GROUP xmlns="http://www.w3.org/2002/01/P3Pv1">
      <EXTENSION>
      <p3p11:data-group>
        <p3p11:datatype>
          <p3p11:dynamic>
            <p3p11:clickstream>
              <p3p11:other.httpmethod/>
            </p3p11:clickstream>
          </p3p11:dynamic>
        </p3p11:datatype>
      </p3p11:data-group>
    </EXTENSION>
      <DATA ref="#dynamic.clickstream.other.httpmethod"/>
    </DATA-GROUP>
  </STATEMENT>
 </POLICY>
</POLICIES>
```

Appendix 3 Problems with ATXP

This appendix describes all the problems of ATXP found in my thesis research.

3.1 Misunderstandings of XACML and XPath Syntaxes

On page 10 of [10], the identifier "subject:subject-category" is not specified in either XACML or the project documentation. The XPath used to select the category of Subject-access element in <Record> elements with "ID" as its id attribute value is wrong. The XACML code fragment which determines if the person identified by his or her subject-id is the owner of the record requested, and if the category of the <Subject-access> element in the requested id record is equal to "contact" (case 3) should be something like the following.

```
<Condition>
 <Apply FunctionId="and">
  <Apply FunctionId="string-equal">
   <Apply FunctionId="string-one-and-only">
    <SubjectAttributeDesignator AttributeId="subject-id"/>
   </Apply>
   <Apply FunctionId="string-one-and-only">
    <AttributeSelector RequestContextPath="//Records/@subject-id"/>
   </Apply>
  </Apply>
  <Apply FunctionId="string-equal">
   <AttributeValue>contact</AttributeValue>
   <Apply FunctionId="string-one-and-only">
    <AttributeSelector RequestContextPath="//Records/Record[@id='ID']/Labels/Subject-access@category">
   </Apply>
  </Apply>
 </Apply>
</Condition>
```

3.2 Ambiguities of Mapping

1) <P3P:POLICIES>: The original APEX mapping transforms all the XACML <PolicySet> elements to the corresponding P3P <POLICIES> elements. However, only the relevant <PolicySet> elements which include at least a rule applicable to the customer private data records are mapped to the corresponding P3P <POLICIES> elements. The bolded code fragment which checks whether the resource requested is a customer private data record should appear in the <Target> element or the contained rules of a relevant <PolicySet> element.

[code fragment 1]

```
<PolicySet PolicyCombiningAlgId=""permit-overrides>
```

```
<Target>
.....
</Target>
<Policy>
 <Rule>
  <Target>
   <Resources>
    <Resource>
     <ResourceMatch MatchId="string-equal">
      <AttributeValue> urn:example:schemas:records</AttributeValue>
      <ResourceAttributeDesignator AttributeId="resource:target-namespace"/>
     </ResourceMatch>
    </Resource>
   </Resources>
  </Target>
 </Rule>
</Policy>
..policy 2
.....
</PolicySet>
```

2) <P3P:EXPIRY>: In ATXP, the mapping checks the XACML code which verifies the request time with the designated time, but the code fragment itself is wrong. The correct code fragment is the bolded part of code fragment 2. For each relevant XACML <PolicySet> element which contains code fragment 1, the contained applicable rules which has the code fragment 2 are checked, and the designated dates are chosen (i.e. 05-01-2004 in code fragment 2). Among the chosen dates, the most recent date should be selected to generate the <EXPIRY> element of the corresponding P3P <POLICIES> element.

[code fragment 2]

```
<PolicySet PolicyCombiningAlgId=""permit-overrides>
 <Target>
 .....
 </Target>
 <Policy>
  <Rule>
   <Condition>
    <Apply FunctionId="date-less-or-equal">
     <Apply FunctionId="date-one-and-only">
      <EnvironmentAttributeDesignator AttributeId="environment:current-date"/>
     </Apply>
     <AttributeValue>05-01-2004</AttributeValue>
    </Apply>
   </Condition>
  </Rule>
 </Policy>
 ..policy 2
 .....
</PolicySet>
```

3) <P3P:POLICY>: The ATXP mapping does not specify the eligibility of XACML <Policy> elements. A <Policy> may not have a <Target> element explicitly, but it can inherit it from the outer <PolicySet> element or inner <Rule> elements. Therefore, a <Policy> element whose <Target> element or inherited <Target> element checks whether the requested resource is a customer private data record should be checked and transformed if applicable. Another problem the the mapping is that the code fragment checking the resource id is included in the pattern. Such code is unnecessary since it is very unlikely that an e-business stores all the customer data in a single data file.

4) <P3P:STATEMENT>: In the mapping pattern, the code which checks whether the request issuer is "nonowner" is wrong. The code should check whether the "subject-id" is not the same as the owner id stored in the requested record instead of "Subject-category" which is neither specified in XACML specification nor the project documentation.

5) <P3P:CONSEQUENCE>: The ATXP mapping does not clearly state the eligibility of the chosen <Description> elements. It should state that only the rules which grant read access to customer records for "nonowner" are selected and their <Description> elements are mapped to the P3P <CONSEQUENCE> element contained in their corresponding P3P <STATEMENT> elements.

6) <P3P:PURPOSE>: The attribute identifiers "action:action-purpose" and "action:action-specific" are specified neither in the project documentation nor in the XACML specification. The task-purpose attribute of the Record element has value of "admin" but the generated contained element of P3P <PURPOSE> element is <current/>. Another problem with the mapping is information for generating P3P <PURPOSE> element can be directly extracted from the customer private data record instead of the code pattern provided. XACML is usually used to specify access control policies based on RBAC. The reason why XACML rules need to check the action purpose and whether action purpose is related to requester's role are not explained in the project document at all. Moreover, the data records contain the private information already collected from the customer; it is meaningless to have an XACML access control policy rule which checks the value of the "opt-in" or "opt-out" attributes.

7) <P3P:RECIPIENT>: The code which checks a requester's organization domain should be included in a <SubjectMatch> element of the <Target> element of an applicable rule. In the example shown in section 4.1.1 of XACML 2.0 specification, the code should check whether the request's "subject-id" attribute (i.e. the request's email name in the company domain) is the same as the company domain, the delivery company domain, or a company which follows equivelent practices. Thus the mapping should extract information from the XACML code described above to generate the P3P <RECIPIENT> element. If the information of the recipients is available in the data record of a certain type (e.g. ID type) then it can be directly extracted from the data

record instead of from the applicable XACML policies. There are two other minor problems with the mapping. The first problem is attribute identifier "subject-category:subject-recipient" is neither specified in the XAML specification nor explained in the document. The second one is that it is meaningless to have XACML code which checks the attribute "opt-out" or "opt-in", since the requested data record is already collected and stored from customer.

8) <P3P:NON-IDENTIFIABLE>: The meaning of the P3P < NON-IDENTIFIABLE > element is that all the data collected under a <STATEMENT> element is anonymized. Such information should not be stored in records which have customer id attached (e.g. 3.2.1.1). If a company collects anonymized data, it is very likely that it has a separate file for storing such data. The code which the maps applicable rules should check if the requested resource is this file.

Appendix 4 Problems of ATPX

This appendix describes all the problems of ATPX found in my thesis research.

4.1 The problems caused by misunderstanding of P3P

On page 8 of [20], the example of a P3P privacy statement and the corresponding rule in XACML is wrong. P3P policy only states the categories of data that will be collected and stored by the website. Thus, it does not mean to write collected users' private data from HTTP access log to the source file of the website's homepage. The corresponding rule in XACML should state that only a user's clientAddress (i.e. user's IP address) and referrer included in the http request from user, the userName (i.e. the user's id on the website probably stored in cookie), and the time when the http request sent by user are recorded in the HTTP access log or other data storage (e.g. the database).

The medical record example on page 8 of [20] is also wrong. Privacy statement "protected health information must be disclosed only to an identifiable individual or the individual's personal representative" cannot be expressed in P3P since only generic information such as the categories of data collected, the generic recipient and the purposes of data collection. The most closest privacy statement can possibly be expressed in P3P is "user health records are collected and stored in the clinic and will be used for curing purpose in the clinic only". It seems that this inappropriate example was taken from section 4.2.4.2 of XACML specification 1.0 [3].

4.2 The problems caused by ignoring APPEL

On page 9 of [20], users should express their privacy preferences in A P3P Exchange Language (APPEL) instead of in P3P policies; The P3P enabled user agent should (e.g. web browser) evaluate only the P3P policies of the website being visited against the user privacy preferences instead of the P3P policies of any website that the user might access.

4.3 The problem of negotiation in case of privacy practices contradiction

If the user privacy preferences contradict the website's P3P policies, the resulting P3P policy through negotiation must be a concessive policy based on the website's P3P policy with the acknowledgement to the user preferences since the user is the owner of the data that the website is about to collect (or collecting). The resulting P3P policy cannot be mutually satisfying, since either the user or the website must concede to reach a common agreement on privacy practices regarding the user private data. Nowadays, users have many choices over the online service

providers that provide similar services. In order to prevent losing potential customers, the service provider (i.e. the enterprise or the website) should be the party to concede.

4.4 The problems of the mapping implementation

1 <P3P:ACCESS>
On page [13] of [20], the code fragment that checks whether the requester is the owner of the private data requested is wrong. "string-one-and-only" is a function id which cannot be used as a attribute id. "SubjectCategory" may be used as the attribute id instead, but its value "access-subject" means the requester is the entity that initiated the access request. This code fragment cannot determine if the requester is the private data owner. Based on the assumptions that a user's identifiable data is stored in a XML file named "Records", and the <Records> has a attribute called "subject-id" whose value is the user's name. The correct code which determine if the requester is the owner of the private data is shown as follows:

```
<Condition>
 <Apply FunctionId="string-equal">
  <Apply FunctionId="string-one-and-only">
   <SubjectAttributeDesignator AttributeId="subject-id"/>
  </Apply>
  <Apply FunctionId="string-one-and-only">
   <AttributeSelector RequestContextPath="//Records/@subject-id"/>
  </Apply>
 </Apply>
</Condition>
```

On page 13 and 14 of [20], the XACML fragments for the 3 cases are wrong. The attribute "resource-location" is used to denote the types of identifiable data that the user has access to, but the connection between the location of the resource and the type of identifiable data is explained neither in [20] nor in the XACML specification. The most appropriate standard attribute in this situation should be "urn:oasis:names:tc:xacml:1.0:resource:resource-id". For example, given a group of private data stored in a <Record> element with the value of the attribute record-id equal to "contact" in the XML file "Records" for user "Paul", then the attribute value "contact" can be used to determine the type of private data requested.

2 <P3P:RECIPIENT>
As described in the problem of mapping <P3P:ACCESS> above, the attribute "subject-category" and the attribute value "access-subject" cannot be used to determine the organization domain of the requester in terms of the recipient group involved. Moreover, the "string-one-and-only" is a function id, thus cannot be used as an attribute id. The following code fragment can be used to determine the organization domain.

```
<Target>
 <Subjects>
  <Subject>
   <SubjectMatch
   MatchId="urn:oasis:names:tc:xacml:1.0:function:rfc822Name-match">
    <AttributeValue
    DataType="http://www.w3.org/2001/XMLSchema#string">
    www.automart.com
    </AttributeValue>
    <SubjectAttributeDesignator
    AttributeId="urn:oasis:names:tc:xacml:1.0:subject:subject-id"
    DataType="urn:oasis:names:tc:xacml:1.0:data-type:rfc822Name"/>
   </SubjectMatch>
  </Subject>
 </Subjects>
</Target>
```

However, the code fragment above itself cannot completely address the problem since the connections between third party requesters and recipient categories cannot be expressed in the P3P <RECIPIENT> element. Thus the resulting XACML policy does not contain the code to check the requester's organization domain in terms of the recipient category involved. This problem is solved in section 4 by the proposed Privacy Control Extension of P3P.

3) Problem of "read" only access

As private data of users is collected by a website, each user is the owner of the data collected by the website. The data owners should be allowed to perform inserting, updating or deleting on their own private data record stored by the organization when necessary. For example, when a user moves to a new address, he need to update the postal address of his online profile on the website to maintain the service (e.g. purchase online) provided by the organization. Note that the data owners should not be allowed to have direct access to the storage of private data collected due to security concerns. However the web site servlet can be used as the agent of data owners on the server side to perform such write access. The categories of private data collected to which users should have write access are expressed in the P3P <ACCESS> element at <POLICY> level. Such information however is not captured by the Automated Privacy Policy Mapping in [20]. This problem is addressed in section 4.

4) Problem of <AnySubject/>

The entities which are allowed to access the users' private data collected by a website are either the data owner, the employee of the organization, or the employee of the organizations which are in the recipient categories specified in the P3P <RECIPIENT> element. The ownership of private data being requested can only be checked in XACML <Condition> element since both requester's id and the owner id stored in the private data record being requested are required. However, the identity check for the two groups of employees (i.e. 2nd party and 3rd party) can be done in the <Subjects> of the<Target> of the resulting <Policy>. In addition, the XACML policies derived by mapping P3P regarding user private data are used together with other

XACML access control policies regarding other organization resources by the access control mechanism. Therefore the identity check for the two groups of employees should be done in the <Subjects> of the <Target> of the resulting <Policy> in order to achieve faster policy indexing.

5) Useful P3P elements are not covered by the Automated Privacy Policy Mapping
Information contained in the <ENTITY>, <PURPOSE>, and <RETENTION> are essential to achieve privacy control enforcement. The email address of the organization contained in <ENTITY> can be used to check request subject's id. For example, an organization has an mail address "service@automart.com". In the <Condition> element a function can easily be implemented to check if the value of request subject's id matches the pattern "*@automart.com" to determine if the request subject is an employee of the enterprise.

The intended uses of private data collected are expressed in the <PURPOSE>. Such information should be captured and transformed somehow to the corresponding RBAC model of the organization that can be integrated into the resulting XACML access control policies to enforce privacy control. For example, an organization states in its P3P policy that users' postal address information is collected and used to complete users' online orders only. But the XACML polices derived by using the Automated Privacy Policy Mapping will check only the organization domain of the request subject thus may allowing a marketing division employee to access users' postal address information. This is an obvious privacy violation not prevented by the resulting XACML policies derived by using the Automated Privacy Policy Mapping. This problem is addressed in Section 4 of my thesis where the proposed Privacy Control Extension of P3P covers the role-purpose binding.

Although the time period for which the private information is retained cannot be specifically stated directly in the <RETENTION> element except the extreme cases (i.e. <no-retention/> and <indefinitely/>), it can be specified in the website's human-readable privacy policy. Moreover, a link to the human-readable privacy policy is required to be included in <RETENTION> element. Hence, if the retention information can be extracted somehow from the human-readable policy or is specified in a pre-defined file, it can be used with the P3P <RETENTION> element together to create a Rule Condition that check the time of request against the pre-determined time. The mapping details are described in section 4.

6) Aggregating <STATEMENT> elements
As it is advocated in [22] and [20], the <STATEMENT> elements which are specified regarding the related or identical private data should be aggregated into a single <STATEMENT> element so that the efficiency of resulting XACML policies would be improved. This is not always applicable since identical private data is seldom collected on multiple parts of a website thus there is no need for aggregating the related <STATEMENT> elements. Moreover it is not always the case that related private data to collect can be covered by a single <STAEMENT>.

For example, on the checkout page of a online store, users are required to submit address and credit card information. The address information collected can be used for marketing purpose and disseminated to other party if the user agrees with it. The credit card information however must be used for <current/> purpose only and must not be disseminated to other party. Aggregating the two pieces of related information with the same group of <RECIPIENT> elements and <PURPOSE> element in to a single <STATEMENT> element will probably scare away the potential customers.