Accountability and Enforceability of Enterprise Privacy Policies

Yolanta Beres, Pete Bramhall, Marco Casassa Mont,
Mickey Gittler, Siani Pearson
Trusted Systems Laboratory
HP Laboratories Bristol
HPL-2003-119
June 30th , 2003*

E-mail: yolanta_beres@hp.com, pete_bramhall@hp.com, marco_casassa-mont@hp.com, mickey_gittler@hp.com, siani_pearson@hp.com

privacy policy, privacy language, accountability, enforcement, mobility

This paper describes our approach to the evolution of enterprise privacy policies and related privacy management ecosystems. We argue that it is important to be able to express enforceable privacy policies, to explicitly manage accountability and to develop the whole privacy infrastructure, rather than just one part of this. In this paper we briefly illustrate our philosophy and vision, present a list of key requirements and describe our ongoing research. In our approach we emphasise three key aspects: extension of policy languages to allow specification of the use of trust and security techniques, enforceability of such privacy languages and management of accountability across enterprises in respect of privacy policy enforcement.

* Internal Accession Date Only
Approved for External Publication

Accountability and Enforceability of Enterprise Privacy Policies

Yolanta Beres (yolanta_beres@hp.com)
Pete Bramhall (pete_bramhall@hp.com)
Marco Casassa Mont (marco_casassa-mont@hp.com)
Mickey Gittler (mickey_gittler@hp.com)
Siani Pearson (siani_pearson@hp.com)

Trusted Systems Laboratory (TSL)
Hewlett-Packard Laboratories, Bristol, UK

Abstract

This paper describes our approach to the evolution of enterprise privacy policies and related privacy management ecosystems. We argue that it is important to be able to express enforceable privacy policies, to explicitly manage accountability and to develop the whole privacy infrastructure, rather than just one part of this. In this paper we briefly illustrate our philosophy and vision, present a list of key requirements and describe our ongoing research. In our approach we emphasise three key aspects: extension of policy languages to allow specification of the use of trust and security techniques, enforceability of such privacy languages and management of accountability across enterprises in respect of privacy policy enforcement.

Keywords: privacy policy, privacy language, enforcement, accountability, and mobility

1. Introduction

Personal identities and profiles are becoming more important as they enable interactions and transactions in inter-enterprise environments, on the web, in federated e-commerce sites and in mobile environments.

Digital identities and profiles enable business and social opportunities among people and organisations, including public administrations; on-line communication and the related easier gathering and transmission of information give rise to a number of threats to people's privacy, such as identity and profile thefts, as well as unauthorised disclosure of confidential data.

Privacy management issues are of increasing relevance: these need to be analysed and addressed in a holistic way that still focuses on people and the protection of people's privacy.

In this paper we describe approaches to privacy policy specification and enforcement that potentially could enable citizens and consumers to participate more confidently in the digital economy, and open up more markets for e-commerce and e-government solutions. In particular, new scenarios may be possible that are just too risky today, such as accessing very sensitive personal or corporate information in generally untrustworthy environments.

2. Current Work, Issues and Requirements

Many solutions and approaches have been proposed to address privacy management and the associated trust issues. These include: branding, seals, certifications, self-profiling, periodic audits and legislation acts. People's faith and trust form a basis for such approaches.

In general, technology users have little understanding or knowledge of the privacy laws and legislation that regulate the management of personal information and their implications. Privacy and data protection laws that regulate this area do exist, but they are hard to enforce and monitor, especially when private information is spread across organisations and nations' boundaries. In addition, further complexity arises due to the fact that privacy laws can differ quite substantially depending on national and geographical aspects. For example, within the European Union (EU), people can consent to have their personally identifiable information used for commercial purposes, but the default is to protect that information and to not allow it to be used indiscriminately for marketing purposes.

We believe that creation of the Platform for Privacy Preferences (P3P) [1] was an important step in attempting to standardise how personal information should be disclosed across different platforms and distinct regulation environments. The initial goal of P3P was to create an underlying framework that allows users and web site owners to negotiate what personally identifiable information a user is willing to disclose in exchange for specific types of access to a Web site. The main technical challenge is to do this in such a way as to automate the negotiation for the server at the same time as providing a useful dialogue with each user for which negotiation is appropriate.

The limitation of this approach, however, is that it addresses point-to-point privacy, mainly for web sites. Since user-based privacy negotiations are inherent in this model, it can prove problematic for a user to negotiate the amount of personally identifiable information he or she is willing to divulge if, for example, the user depends on services provided by the Web site owner. In many cases the users could be forced to give up whatever private information the Web site operator asks for in order to gain access to the site's contents.

When multiple parties are involved in obtaining and consequently managing personal information, such as in federated e-commerce or B2B scenarios, the P3P model provides very little assurance that the user's privacy preferences are maintained. We believe that in order for P3P to continue to be seen as a standard for users' privacy protection, it needs to evolve and integrate with other types of privacy-enhancing techniques, and the outcome requires subsequent standardisation.

Nowadays arguably the biggest problem is that users lack control over their personal information, especially after initial disclosure. Third parties also lack control over the confidential information they manage on behalf of their customers, in particular when they disclose it to other organisations, or during transactions or interactions. Without privacy policy enforceability across multiple domains, everything is a matter of trust. Legislation could help in that respect but only in those countries and contexts where it can be applied and where people's rights are respected.

Recent research in the privacy area attempts to address this issue mainly by trying to define privacy policy languages that could potentially be enforced at different points of technology infrastructure, such as application level, or the Internet Protocol (IP) layer, in cookie management software or anonymity-rendering Web surfing proxies. Many policy languages have been developed that contain complex conditional rules to regulate the lifecycle and usage of private data (e.g. the Antigone project [16], PolicyMaker [17], KeyNote [18] and dedicated policy objects [15]).

Emergence of policy languages specifically targeted at privacy protection such as the Enterprise Privacy Authorization Language (EPAL) [2] recently proposed by IBM is a major step in the direction of providing for more control to the users on how their personal information should be managed. These languages allow a fine-grained definition of privacy policies, including positive and negative rights, obligations, conditions, etc.

An outstanding issue still exists with regard to how the policies can be enforced. For example, how does the user know that the recipient will abide by the policies, and so on, down the chain as the material is forwarded on? Left unanswered also are questions of how to make enterprises and organisations accountable and how to prevent accidental or malicious leakage of confidential information. In essence, in addition to the development of languages such as EPAL, there is a need for technology to address lack of control over the destiny of private data by both users and enterprises, with particular regard to multiparty interactions.

The recent Microsoft's DRM services proposal [11] can be seen as a step in the direction of policy enforceability; it advocates being able to control how data is forwarded, printed and so on, via an application-based approach. Related work also includes Simone Fischer-Hübner's formal privacy policy and its implementation in Linux security [3,4], which uses access control and task-related context information to reason about privacy.

In the next sections of this paper we argue that privacy enforcement mechanisms should be application-independent and allow for greater choice and flexibility. The policy languages should be able to reason not just about subjects and their identities, but about machines and devices and their identities, as well as about requirements placed by different execution environments. To ensure that privacy policies are applied to the personal data at all times, even if the data is moved around across systems and organisations, the mechanisms are also necessary to strongly associate privacy policies to the data; any attempt to remove the policy should be ease to detect and block.

2.1 Emerging Privacy Issues in Mobile Environments

Mobility introduces specific privacy and trust challenges, because of the heterogeneity of the complex environments where mobile appliances are used for digital interactions and transactions. Mobile devices host various profiles used to configure the devices or capture user-specific information; as such, the profiles contain sensitive and private data. The profiles can be distributed amongst several devices that collaborate under functional and social criteria. The distributed information has to be kept coherent and private [19]. Current security mechanisms cannot totally guarantee the privacy of information stored on these devices, especially as their physical location and functional characteristics are changing, leading to a dynamic context: environment and state. The user needs a mechanism to define and enforce policies applicable to the dynamic placement of sensitive and private information. For this, the user needs some indicator of the trustworthiness of the appliance device (via a confidence level, computed for trust rating). In Section 4.1 we discuss how this information can be communicated through privacy policies.
It is also important to "sense" and to gather information about the surrounding environment in a focused manner, in order to capture information specific to the overall business application. The resulting trust measurement can be used as a constraint specified within the privacy policies, for example to restrict viewing of sensitive material in potentially untrustworthy environments, such as when using public transport. If possible, such constraints should also take account of the device owner's judgments of the involved risks and threats, in order to achieve a balance between awareness and transparency to software applications.

2.2 Requirements for Privacy Management

The management of privacy is complex: it involves multiple aspects, including privacy languages, enforcement mechanisms and end-user solutions.

The following non-exhaustive list includes key high-level requirements for privacy management:

- Extensible and flexible privacy languages which can adapt to changing legislative and technical environments;
- "Strong association" of privacy constraints to confidential data - ideally this association should be tamper resistant;
- Strong and provable enforcement of privacy languages, in a variety of contexts, at the appropriate level of abstraction;
- Objective measurement and judgement of the "trustworthiness and appropriateness" of the IT platforms and solutions used by involved parties when dealing with confidential data;
- Provision of a range of options or behaviours, depending upon what such metrics are;
- Explicit management of enterprise accountability when dealing with privacy aspects, including feedback or alarms;
- Direct and active involvement of data owners (or TTPs), if desired or required;
- Simplicity, ease of use and transparency for the user.

3. Our Approach

We recognise the important need to deploy privacy languages in a variety of contexts. In our view, privacy languages not only need to be rich and flexible, but also need to be enforceable, at the right level of abstraction. They are an important part of the development of the whole privacy infrastructure, and not just one part alone.

Because of the complexity and heterogeneity of the involved IT solutions, systems and related infrastructures, we believe that privacy enforcement, in most cases, cannot be achieved only at the "application" level. Depending on the risks and importance of confidential or private data, enforcement needs to happen at the most appropriate level or levels, including platform and OS levels.

In addition we argue that accountability is a fundamental aspect in dealing with privacy management and needs to be explicitly managed to mitigate risks and increase trust in the system. Despite the efforts being made by organisations in dealing with privacy, it is very unlikely that "tamper proof" solutions will ever be provided, considering that solutions are the result of a compromise between business needs, costs and requirements. In this respect, we believe in the importance of active tracing of disclosures of confidential data and auditing by trusted third parties to collect tamper-resistance evidence about enterprise activities when dealing with confidential data.

In the next sections we briefly describe our ongoing work in the privacy area including accountability-assuring privacy models, multi-level enforcement mechanisms and applications in mobile environments. The goal of this is to show that different types of privacy enforcement mechanisms place requirements on what privacy policy languages should be able to express. We believe that work on privacy policy languages has to move in the direction of being able to deal with the constructs pertaining to different execution and enforcement environments that could potentially be deployed for privacy enforcement across organisational boundaries.

4. Expressiveness of Privacy Policies and Accountability

A key issue is the enforcement of privacy once confidential information is parted from its owner and "disseminated" across boundaries during interactions and transactions. Mechanisms and solutions are required that increase accountability of enterprises when dealing with confidential information, at the same time protecting people's privacy and giving people more control over their personal information. The privacy model that we are developing addresses these issues with the following properties:

- **Strong association of privacy policies to confidential data**: association of "tamper resistant" sticky policies (defining privacy constraints, authorizations, obligations, etc.) to data, defined by users or trusted third parties acting on their behalf;
- **Enforcement of confidentiality**: Usage of sticky policies to obfuscate confidential (personal) information until conditions to allow its disclosure are met. Any tampering with these policies will prevent the access to the content of obfuscated data;
- **Policy compliance checked by trusted third parties**: any disclosure of confidential data is subordinated to the fulfilment of sticky policies' constraints. This is checked and enforced by trusted third parties (TTPs), here referred to as trust authorities. One or more TTPs can be involved in the process;
- **Accountability management**: auditing and tracing disclosure of confidential data via TTPs;
- **User involvement**: active involvement of users (if desired) during the process of disclosing confidential data.

Our privacy model [20] allows data owners (or third parties acting on their behalf) to strongly associate privacy policies to any aggregation of their data before exposing it to third parties (such as e-commerce sites, enterprises, etc.). Once the privacy policy has been defined, it is bind to the personal data through encryption, at the same time obfuscating the data. These privacy policies cannot be parted from the personal data they are associated to or tampered with. The model requires that one or more TTPs be involved in the process. These TTPs check for the compliance to privacy policies before disclosing the personal data to other external parties. Only in positive cases they will allow for the de-obfuscation of the confidential data. Potentially, data owners might run their own TTPs.

The next two sections provide more details about the peculiarities of our privacy policies and an implementation of our model to enforce accountability.

4.1 Privacy Policies

An important aspect of our model is the usage of flexible privacy policies. The aim is for privacy policies to be able to describe different types of privacy constraints that are refinable and enforceable at different levels of abstraction: service, application, platform and operating system (OS). At the current stage we are not so concerned about the specific format of the language that would be used to express these policies; what matters is the possibility to express a flexible set of conditions and constraints.

Current privacy policy languages like EPAL [2,5,6] cannot adequately deal with the constraints at different levels of abstraction, and therefore we believe there is a need for such languages to be extended. The constructs that should be present include references to logical names of identity and profile attribute(s); personal data disclosure constraints; obligations; requirements; actions, such as notification of the data owner in case of multiparty disclosure

or requests for explicit authorization; expiration dates, etc. On the other hand, important constructs are also ability to reason about the state of platform where the data is being transferred, the type of device and environment where the data is being viewed. For example, we could have two different privacy constraints: (a) "non-sensitive data can be sent out across the network; sensitive data can only be sent out to partners who are certified by a trusted third party (TTP) or else must be encrypted to at least a given level and sent only to trusted platforms" and (b) "data can only be displayed on a given device used by a given user if it is either not sensitive or else the device is a trusted platform and the environment has a trust rating of at least 3 and the user fulfils the role of a customer relations member". These constraints can be expressed in the following more formal format:

(a): *can_do(send(Data, Remote_device)):-*
¬is_sensitive(Data) ∨ *(trusted_platform(Remote_device) & encrypted(Data, min_level))* ∨
owner_cert(Remote_device)

(b): *can_do(display(Data, Current_device, User)):-*
¬is_sensitive(Data) ∨ *(trusted_platform(Current_device) & env_trust_level(3) &*
role(User, customer_relations))

In both cases the enforcement happens on the data owner's device that has to decide whether the privacy constraints are met before the data is sent or displayed. Another possibility is for checks on whether the remote platform meets the constraints to be carried by a TTP. In such cases the constraints can be included within a public encryption key. High-level policies can be defined to contain constraints and actions that can then be refined at different levels of abstraction. As an example, the following policy not only specifies constraints that the TTP could verify but it also embeds a policy of a similar form to those above with the exception that it can be deployed and enforced at the TTP's side and at the receiver's side rather than at the sender's or original data owner's side.

<sticky policy> // privacy policy
<attribute> // *name of the attribute*
Data
</attribute>
<owner>
//reference name – encryption key
<reference name> *pseudonym1* </reference name>
//encrypted owner's call back address
<owner's details>
encrypted call back address
<owner's details>
</owner>
<validity>
expiration date
</validity>
<action>
notify_owner_before_disclosure
</action>
<constraint> // constraint that can be easily checked by TTP
X.509_authentication_required & receiver.DN != ("ACME.com")
</constraint>
< constraint> // constraint targeted to check the potential trustworthiness of remote platform or its owner before
// permitting decrypt ion of the data on the remote platform
can_do(read(Data, Remote_device)):-
(*trusted_platform(Remote_device) & trusted_state(Remote_device))* ∨ *owner_cert(Remote_device)*
</constraint>
</sticky policy>

These kinds of policies are important from a privacy perspective as they express users' preferences about how sensitive data should be treated when sent across a public channel and about the kinds of devices and environments in which the data can be used. If the languages are able to deal with these types of constraints, the mechanisms described in the next sections then can be used to enforce them at the right level of abstraction, including trusted platforms and operating systems.

4.2 Accountability

Two enforcement aspects are important in our model: the enforcement of privacy polices and the enforcement of accountability.

It is important to ensure that privacy policies, associated to confidential data, are enforced: it must not be possible to tampering with them or parting them from this data. However, despite efforts and good will, it might occur that unauthorised disclosures happen especially if the enforcement is at the application level.

Auditing mechanisms need to be put in place to enforce accountability of the parties that deal with confidential data. These mechanisms need to collect (undeniable) evidence about usages of this data and the fulfilment of privacy policies. This data will be used to track misbehaviours and for forensic analysis.

We are exploring the usage of Identifier Based Encryption (IBE) technology [7,8,9] as a potential mechanism for ensuring that privacy policies "stick" to the data as it is transferred from data owners to other involved parties. We are investigating how privacy policies can be used directly as IBE encryption keys to obfuscate confidential data. To achieve this we are exploiting the following core properties of IBE technology: ability to use any kind of string (i.e. sequence of bytes) as an IBE encryption key (publicly available); possibility to postpone in time the generation of IBE decryption key; reliance on at least a trusted third party, called in this approach a trust authority (TA), for the generation of IBE decryption keys.

Figure 1 shows the architecture of a system based on our privacy model. More details can be found in [20]. Messaging protocols (1)-(4) are carried out in order, and involve transfer of the information indicated in the directions shown by the arrows.

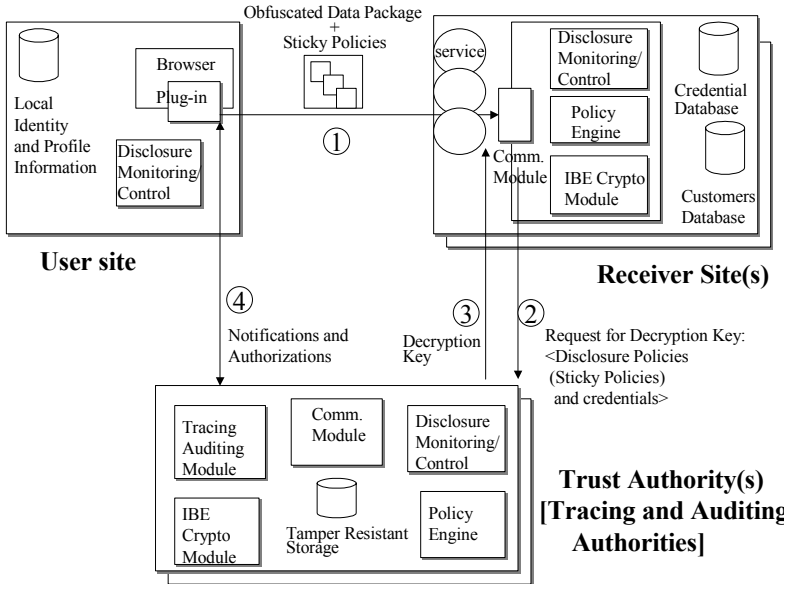

Fig. 1. High Level Architecture

Identity or profile information is encrypted with "sticky policies" (1), before its disclosure to third parties, by web browser plug-ins or trusted applications.

These policies are used as IBE encryption keys and might include: references to logical names of identity and profile attribute(s); disclosure constraints; obligations; actions (i.e. notification of the owner in case of multiparty disclosure); expiration dates, etc.

To obtain a valid IBE decryption key (2), the receiver needs to interact with trust authorities (TAs, i.e. TTPs) and provide information (including authentication credentials, business related information, company/individual policy related to data disclosure, usage and storage, software state, platform configuration etc.) as required by the disclosure policies. In doing this, the receiver is explicitly aware of (and understands) these policies.

A TA will issue an IBE decryption key (4) if it acknowledges the compliance with the disclosure (sticky) policies. Before doing this it might interact with the information owner (3) to ask for his/her authorization or to send a notification. The TA traces and stores all the information exchanged during these interactions in audit-trails, as evidence for future contentions or forensic analysis.

Users' identity and profile information is exchanged by means of data packages containing encrypted data and their associated sticky policies.

Use of a TA in this model introduces more accountability for the involved parties as the TA can trace and store all the information exchanged during the interactions in audit-trails, as evidence for future contentions or forensic analysis.

The usage of IBE technology is just a possible option. Similarly, other alternative public key cryptography based approaches can be potentially used here.

The described model can be easily deployed at the application and service levels. We currently have working implementations of most of the involved components. It provides "best effort" mechanisms to enforce privacy and underpin accountability. Nevertheless, once data is disclosed and it is in clear, at the enterprise site, it can still be misused. It can leak for accidental or malicious reasons. This introduces the importance of having "stronger" privacy enforcement mechanisms, at the right level of abstraction, including trusted platforms and operating systems. The next sections provide more details about these mechanisms.

5. Enforcement

Having extensive privacy policies in an enterprise does not directly ensure privacy protection if there are no effective means of consistent policy enforcement across multiple applications and across enterprise boundaries. Once a sufficiently rich policy language is used and adopted, resulting systems are greatly strengthened if the policies that govern the use of data can be enforced at the right level of abstraction. In the previous section we described a possible approach to enforce privacy polices at the application and service level. This might not be sufficient. What is also needed is an end-to-end privacy policy enforcement framework that cannot be easily circumvented through accidental errors or malicious actions.

Depending on the risk and importance of the confidential data, stronger enforcement mechanisms might need to be deployed at the right level of abstraction of the IT stack. In this section we describe two potential solutions at the platform layer. The first one uses the features provided by trusted hardware components; the second one describes mechanisms introduced at the operating system level for enforcement of policies that require precise control across multiple applications. It is probably necessary to still build on top of these enforcement mechanisms so that privacy enforcement is done over multiple levels of abstraction, not just at the platform level.

5.1 Enforcement by Trusted Platforms

Increasing privacy issues and emerging e-business opportunities that demand higher levels of confidence have led the Trusted Computing Platform Alliance (TCPA) (and now the Trusted Computing Group (TCG)) to design and develop a specification for computing platforms [12,14] that creates a foundation of trust for software processes, based on a small amount of hardware. An allied technology, called Next-Generation Secure Computing Base (a trusted operating system that can run in parallel to Windows), is currently being developed by Microsoft [10].

A trusted (computing) platform is one that has a trusted component, probably in the form of built-in hardware, which it uses to create a foundation of trust for software processes.

Trusted computing provides the following key features that provide building blocks for privacy:

- *protection for users' secrets*: 'protected storage' functionality binds secrets to a platform and can even prevent the revelation of secrets, unless the software state is approved.
- *potential for remote trust*: users or enterprises can recognise that a platform has known properties and identify that a system will behave as expected.

It is also possible to create trust domains that are based on multiple trusted platforms.

The TCPA specification deliberately minimises both the potential for identifying a platform across multiple uses and the use of identifying data. The TCPA specification deliberately does not include attempts to identify who is making statements or communicating; instead it prefers to allow the use of attributes or credentials and gives the communicating entity better reason to trust these attributes. There is never one stable identity across transactions, thereby avoiding the privacy-related pitfalls of having a unique identity. For further information about the privacy-positive design of TCPA, see [12].

This technology may be extended specifically to enhance privacy, for example, in order to restrict sensitive information that is divulged [13] and to build new identities when they are needed. As illustrated in an earlier section, privacy languages can explicitly contain conditions and obligations to be met by computing platforms. Furthermore, privacy constraints can be verified before data disclosure by data owners, trusted third parties or enterprises and can condition further related transactions and interactions.

In summary, trusted platforms provide building blocks for privacy, but do not provide a complete privacy solution; other mechanisms such as identity management would be needed in addition, including mechanisms to control data flow, during the execution of applications that manipulate confidential data.

5.2 Enforcement by a Tagged OS

Enforcement of privacy policies within an operating system is an important aspect in that it ensures that privacy policies are applied uniformly in the platform independently from any applications deployed on top. A potential approach that we are investigating is to constantly tag the data with a privacy label as the data is manipulated and propagated in a computer system.

In our approach a privacy label can be associated with any arbitrary piece of data, be it a text document or a large digital image file. The label then follows the data throughout the full lifecycle, as the data is manipulated by different applications and transferred between machines and devices, with controls enforced at the operating system level. The "stickiness" of the label to the content, not to the content holder, such as a file, is an important feature as it

ensures that even when the data is copied around, the label follows it as well. For example, a privacy label can be associated with a string of information such as a social security number; when this string is copied to another file the original privacy label follows the string during copying. This feature allows precise tracking of the propagation of labelled data across applications and across systems.

In the operating system each privacy label is directly matched to a set of rules that express requirements for how data with that label can be used. These rules dictate how the labelled data should be handled by applications. For example, the information owner could prohibit the information from being copied, printed or transferred to other machines or devices. The rules are enforced through the policy modules designed as an add-in mechanism within the operating system. This enforcement is invisible to both applications and to users.

Privacy labels enable data to be labeled according to the information relevance to an enterprise, the applicable access control measures, and privacy concerns. The information owners (using standard policy languages, as described previously in section 4.1) should be able to create the rules associated with these labels. These then need to be transformed into an internal format enforceable by the operating system. As applications have no access to the labels and associated rules, the information owners are assured that their data is protected even if computer systems are infected with malicious code, or penetrated by intruders. The advantage of having this type of privacy policy enforcement mechanism in addition to other means lies in that controls apply uniformly across different applications and cannot easily be subverted by them.

6. Discussion

In this document we argued about the importance of being able to express privacy policies that are enforceable across various environments at different levels of computer system: at the operating system or trusted platform level, through the encryption mechanisms such as IBE and through Trusted Third Parties (TTPs), and within applications or services. The enforcement mechanisms potentially dictate what type of conditions and rules a policy language should be able to express. We believe that development of privacy policy languages should be closely linked with applicable enforcement mechanisms. The constructs are necessary that make it possible to express the characteristics of possible execution environments (e.g. mobile or hot-desking), the platform expectations, and operations on different types of data transfer (e.g. printing, displaying and encrypting).

Environments where confidential data is used dictate and influence mechanisms that enforce privacy policies. In this paper we mainly focused on enterprise and e-commerce environments. Further challenges are introduced in more dynamic environments such as in pervasive computing.

Pervasive computing environment needs have fostered a variety of sentient devices that operate with sensitivity to context changes. In many cases, the enforcement of privacy policies has to happen within these devices. Ongoing research is focusing on this problem. The communication or interaction amongst the devices is supported by new context-sensitive middleware software that can sense and analyse context from various sources [15]. This permits private information to be protected in dynamically changing contexts, on the path from one device to the next and to remote applications. Potentially, trusted computing components such as defined by TCPA can be used in these devices to communicate the configuration and integrity metrics of the device. Enforcement and decision engines can combine this information together with other factual data such as state of the surrounding environment, user subjective input and data sensitivity to check compliance with privacy

policies on personal data. Thresholds embedded in the policy rules can then trigger adaptive actions.

The type and strength of enforcement mechanism applicable in an environment (which might be a personal, enterprise or a particular business environment) depends on the level of perceived risk and the sensitivity of the personal data held or disclosed. In our view the strength of guarantee must be sufficient to counter perceived threats. Thus, if the risk level is acceptable or if the threats are completely mitigated by legislation, the mechanisms should exist to reconfigure the privacy policies so that a minimum level of enforcement is performed within the system. However, some type of enforcement mechanism will always be necessary be it by encryption or at the platform level, as legal agreements alone do not ensure that private data will not be disclosed to unauthorized parties. Moreover, most enterprises want to stay within the law, do not want to put effort into understanding what the law means and lack the ability to track changes. Therefore, they are keen to have legal requirements expressed in policies that they can be sure are enforced. Outsourced specialist services can code the law into the policies if necessary, but this will need to be combined with enforcement.

Potential problems could arise when deploying privacy technologies due to the involvement of users in the enforcement loop (for example, they could be unskilled or unwilling to use the technology): mitigation of these problems could be achieved by using TTPs, for example consumer groups trusted by individuals to represent their interests. In addition, legislative aspects and IT complexity can also be a barrier to adoption of privacy-protecting frameworks, and so there is a need for simple, integrated tools for both users and administrators if privacy solutions are to be practical and widespread.

7. Conclusions

In this paper we describe our approach to privacy policy enforcement. This approach addresses three key issues: extension of policy languages to allow specification of the use of trust and security techniques, enforceability of such privacy languages and management of accountability across enterprises with regard to privacy policy enforcement. Work is in progress at HP Labs, in these three areas.

8. References

[1] W3C, The Platform for Privacy Preferences 1.0 (P3P1.0) Specification, September 2001. For updates on P3P and a list of compliant sites, see http://www.w3.org/P3P.

[2] P. Ashley , S. Hada , G. Karjoth, C. Powers, M. Schunter, Enterprise Privacy Authorization Language (EPAL), IBM, 2003.

[3] S. Fischer-Hübner, A. Ott, "From a Formal Privacy Model to its Implementation", Proceedings of the 21st National Information Systems Security Conference, Arlington, VA, October 5-8, 1998, http://www.cs.kau.se/~simone/niss98.pdf

[4] S. Fischer-Hübner, IT-Security and Privacy - Design and Use of Privacy-Enhancing Security Mechanisms, Springer Scientific Publishers, Lecture Notes in Computer Science, LNCS 1958, May 2001, ISBN 3-540-42142-4.

[5] G. Karjoth, M. Hunter, A Privacy Policy Model for Enterprises, IBM Research, Zurich - 15th IEEE Computer Foundations Workshop, June 2002.

[6] G. Karjoth, M. Schunter, M. Waidner, Platform for Enterprise Privacy Practices: Privacy-enabled Management of Customer Data - 2nd Workshop on Privacy Enhancing Technologies, Lecture Notes in Computer Science, Springer Verlag, pp 69-84, 2002.

[7] D. Boneh, M. Franklin, Identity-based Encryption from the Weil Pairing. Crypto 2001, 2001.

[8] C. Cocks, An Identity Based Encryption Scheme based on Quadratic Residues. Communications - Electronics Security Group (CESG), UK. http://www.cesg.gov.uk/technology/id-pkc/media/ciren.pdf, 2001.

[9] L. Chen, K. Harrison, A. Moss, D. Soldera, N.P. Smart, Certification of Public Keys within an Identity Based System, Proc. 5th Int. Information Security Conference (ISC), 2002. LNCS 2433, Springer-Verlag, 2002.

[10] Microsoft Corporation, White Paper on Palladium, June 2002. Available via http://www.microsoft.com/presspass/features/2002/jul02/0724palladiumwp.asp.

[11] Microsoft Corporation, Microsoft Rights Management Solutions for the Enterprise: Persistent Policy Expression and Enforcement for Digital Information, February 2003. Available via http://www.microsoft.com/windowsserver2003/techinfo/overview/rm.mspx.

[12] S. Pearson (ed.), Trusted Computing Platforms, Prentice Hall, 2002.

[13] S. Pearson, A Trusted Mechanism for User Self-Profiling in E-Commerce, Selected Papers from Deception, Fraud, and Trust in Agent Societies workshop, LNAI journal, Springer, 2003.

[14] Trusted Computing Platform Alliance, TCPA Main Specification, Version 1.1, 2001. Available via www.trustedcomputing.org.

[15] S. S. Yau, F. Karim, Y. Wang, B. Wang, and S.K.S. Gupta, Reconfigurable Context-Sensitive Middleware for Pervasive Computing, IEEE Pervasive Computing, July-September, 2002.

[16] The Antigone Project, http://antigone.citi.umich.edu/content/antigone-2.0.11/docs/html/index.html.

[17] M. Blaze, J. Feigenbaum, J. Lacy, Decentralized Trust Management, AT&T Research, Murray Hill, NJ 07974, Proc. IEEE Conference on Security and Privacy, Oakland, Ca, May 1996.

[18] The KeyNote Trust-Management System, Network Working Group, RFC 2704.

[19] S. Riche, G. Brebner, and M. Gittler, Client Side Profile Storage. Presented at the International Workshop on Web Engineering, Pisa, Italy, May 24th 2002.

[20] M. Casassa Mont, S. Pearson, P. Bramhall, Towards Accountable management of Identity and privacy: Sticky Policies and Enforceable Tracing Services, TrustBus 2003 workshop, 2003