A Model-Based Privacy Compliance Checker

Siani Pearson, Damien Allison

HP Laboratories
HPL-2008-193

Keyword(s):
privacy, e-business organization, compliance checking, modeling, governance

Abstract:

Increasingly, e-business organisations are coming under pressure to be compliant to a range of privacy legislation, policies and best practice. There is a clear need for high-level management and administrators to be able to assess in a dynamic, customisable way the degree to which their enterprise complies with these. We outline a solution to this problem in the form of a model-driven automated privacy process analysis and configuration checking system. This system models privacy compliance constraints, automates the assessment of the extent to which a particular computing environment is compliant and generates dashboard-style reports that highlight policy failures. We have developed a prototype that provides this functionality in the context of governance audit; this includes the development of software agents to gather information on-the-fly regarding selected privacy enhancing technologies and other aspects of enterprise system configuration. This approach may also be tailored to enhance the assurance provided by existing governance tools.

External Posting Date: November 21, 2008 [Fulltext] Approved for External Publication
Internal Posting Date: November 21, 2008 [Fulltext]

To be published in International Journal of E-Business Research (IJEBR), special issue on Privacy Technologies 2009.

A Model-Based Privacy Compliance Checker

Siani Pearson* Damien Allison

*Hewlett Packard Research Labs
Filton Road, Stoke Gifford, Bristol, BS34 8QZ, United Kingdom
Siani.Pearson@hp.com

Abstract. Increasingly, e-business organisations are coming under pressure to be compliant to a range of privacy legislation, policies and best practice. There is a clear need for high-level management and administrators to be able to assess in a dynamic, customisable way the degree to which their enterprise complies with these. We outline a solution to this problem in the form of a model-driven automated privacy process analysis and configuration checking system. This system models privacy compliance constraints, automates the assessment of the extent to which a particular computing environment is compliant and generates dashboard-style reports that highlight policy failures. We have developed a prototype that provides this functionality in the context of governance audit; this includes the development of software agents to gather information on-the-fly regarding selected privacy enhancing technologies and other aspects of enterprise system configuration. This approach may also be tailored to enhance the assurance provided by existing governance tools.

Keywords

Privacy, e-business organization, compliance checking, modeling, governance

INTRODUCTION

In order to conduct business, organizations must try to assess and ensure compliance with privacy legislation, policies and regulations, as part of their IT governance initiatives. Such privacy management is an important issue for e-business organizations since e-business can be defined as "the utilization of information and communications technologies (ICT) in support of all the activities of business" (Wikipedia, 2008). This issue involves both operational aspects, related to the enforcement of privacy policies, and compliance aspects related to checking for compliance of these policies to expected business processes and their deployment into the enterprise IT infrastructures.

The Need for Automation

We address the problem of how to make privacy management more effective by introducing more technology and automation into the operation of privacy in e-business organizations. Enterprises are coming under increasing pressure to improve privacy management, both to satisfy customers and to comply with external regulation (Laurant, 2003) or internal policies. Not only are human processes prone to failure but the scale of the problem highlights the desire for additional technology to be part of the solution. The trend towards complexity and dynamism in system configurations heightens this need for automation to ensure that privacy and security properties are maintained as changes occur, and in addition to check that the privacy enhancing technologies are operating as desired.

Automated Compliance Checking Requirements

Most of the technical work done in this space focuses on the provision of auditing and reporting solutions that analyse logged events and check them against privacy policies and process guidelines. These auditing systems usually operate at a low level of abstraction and do not take into account the overall compliance management process that involves both the refinement of privacy laws and guidelines within enterprise contexts, their mapping into the enterprise IT infrastructure and their subsequent checking against the enterprise's operational behaviour.

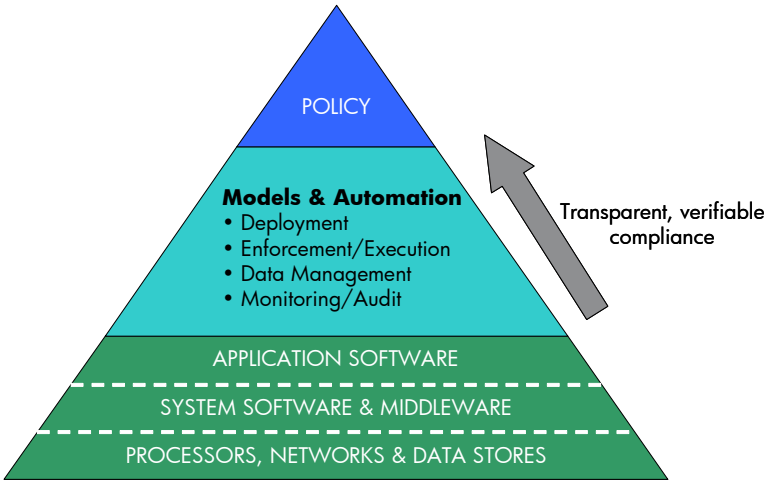

Figure 1. Model-based, policy-driven IT.

At present there is a gap between the definition of high-level regulations, standards and best practices and what is actually happening in an enterprise at the level of application software, system software and middleware, processors, networks and data stores. The current approach is to fill this gap using people-based processes, but there

are drawbacks to this, in terms of being slow, expensive, error-prone and leading to best-effort compliance due to limited resources. Our vision is to bridge this gap where possible with model-based technology and automation, as shown in Figure 1. On the one hand privacy policy enforcement technologies can be used to deliver compliance to privacy principles and goals; on the other hand (the focus of this paper) we can use system monitoring technologies to continuously assess their actual performance and ability to deliver against the requirements of the policy.

Our Approach

To address this problem we are developing a Policy Compliance Checking System. Key requirements of this system are to:

R1. model privacy policies (based on company privacy policies, laws and guidelines or best practice). A mechanism is needed that enables such models to be defined and viewed. Predefined models should also be usable, and amendable by expert users if desired.

R2. *map these models at the IT level.* It is necessary to configure the models to the deployed system.

R3. *analyze related events*. The compliance checking system needs to monitor those properties of the deployed system that can affect satisfaction of the privacy policies.

R4. *generate meaningful reports highlighting compliance aspects and violations*. These reports should be understandable to non-experts, and allow drilling down to a greater level of detail.

This system should supervise and report on the availability of other privacy enhancing technologies (PETs) – for example, privacy policy enforcement systems, obligation management systems and security technologies – and check for inconsistencies on enforced policies, by comparing information coming from different sources.

For example, a privacy related goal an organisation could wish to attain is that personal data is only used for the purposes for which it was collected. This corresponds to a core privacy guideline (Information Commissioner's Office, 2007), A model may be built up which shows how this goal can be satisfied if a logical combination of subgoals may be satisfied. For instance, this goal can be partially satisfied by the subgoal that the organisation uses a technological control that enforces role-based access to data, where roles are associated with processes like marketing or customer support and a check is made such that the data can only be accessed if such processes are included in the allowed purpose of usage for the data. In addition, the system should check that the control is configured correctly, the control is available, the control has not been subverted and there is proper separation of the duties defined for specific roles. So, the goals are mapped to the IT level. In addition to instrumentation and monitoring of the status of the data warehouse to ensure that the data is not accessed in a 'back door' way, by checking system updates, security software, firewall policies, system access and system processes, it would be preferable to instrument the

data client to ensure that it is following best practice policies (with respect to virus, passwords, user access, etc.).

To enhance the decision capability, further subchecks might include for example how the access lists are controlled, who authorises the lists and what training they are given before they enter a username and password. If this technological control were not in place, an alternative method of satisfying the initial goal would be to check process, auditing logs and so on, but this can be very difficult to automate.

POLICY COMPLIANCE CHECKING SYSTEM

This work addresses the problem of explicitly assessing compliance of privacy policies; a similar approach applies to best practice guidelines, legislation and risk analysis. Our system verifies whether the data processing system is strong enough to automatically execute the privacy policies reliably: this involves assessment of the deployment of PETs and the underlying trust, security and IT infrastructure. We aim to allow organizations to check the trustworthiness of their system components, as well as those of their business partners to whom they may transfer personal data. For example, a service may be considered trustworthy if it has been accredited by an independent privacy inspector, (such as BBBOnLine or TRUSTe) (Cavoukian & Crompton, 2000), or a platform may be considered trustworthy if it is judged to be in a trusted state and is compliant with standards produced by the Trusted Computing Group (TCG) (2003).

Overview

This paper describes the system we have developed to allow an organization to assess their policy compliance using a collection of information describing the organizations' resources. Our system will allow the description of a model defining the goals associated with satisfying its policy constraints. Our system can then monitor the organizational resources to verify that the goals described are being satisfied. The specific high level policy models may be templated for specific legislation which can then be analyzed locally and instrumented in specific properties of the modeled system.

With the aim of automating privacy process analysis and configuration checking for enterprises, we use functional decomposition of risks or policies defined by experts up-front, as shown in Figure 2, and then subsequently use this model to dynamically assess systems and generate reports, whenever required.

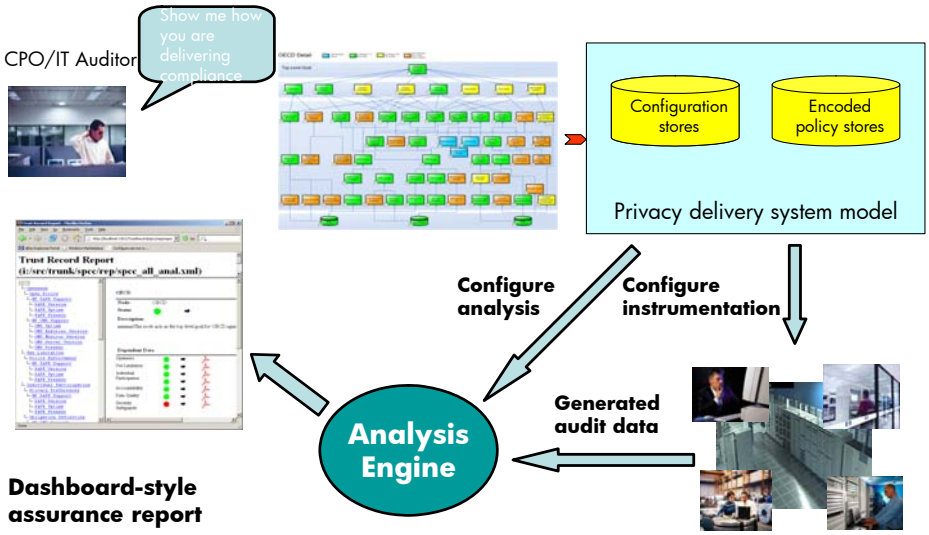

Figure 2. Overview of the system.

The system is intended to be used in the following way: first of all, predefined policy substructures would be input into our editing tool (shown below in Figure 4) by a privacy expert to form a generic privacy model (this only needs doing once, but can be updated subsequently). For each specific enterprise system on which the compliance checker is to be run, a privacy specialist and/or specialised administrator would tune the constraints and deploy the system. Next, agents would be deployed to resources based on information given in the model, and would gather information over a selected time period. Whenever desired, analysis could be triggered and a corresponding report generated (an example is shown below in Figure 5).

Our approach is novel: at the core the compliance checking system works by allowing a policy agent to identify key performance indicators of the live system that reflect attributes associated with goals. The system then monitors and reports in a real-time dashboard. The metrics used are often available in other domains, but have not before been pulled into one place for this purpose; they give the ability to instrument the lowest level checks. Furthermore, we use logical implication to aid policy modelers in investigating and stating what 'compliance' looks like.

Privacy Assurance Models

In order to automate privacy compliance the system assesses the extent to which IT controls (including PETs such as privacy policy enforcement and obligation management systems) satisfy key privacy principles or goals. To do this the system uses a model that allows recursive decomposition of top-level properties down to specific requirements that technology can analyse, enforce and report on. We have already

considered above an example of such technological control influence on a high level goal.

In general, there can be a many-many mapping between the goals and subgoals: for example, it may be necessary to satisfy a combination of subgoals in order to satisfy a higher level goal. At the same time, general subgoals associated with aspects of the system that influence it more widely may affect many supergoals.

The top-level goals can be, as desired, high level specification of privacy principles as suggested for example by Organisation for Economic Co-operation and Development (OECD) (1980) or APEC (Greenleaf, 2005), regulation such as European Directive 95/46/EC on the protection of personal data (European Parliament and Council, 1995) or even specific country legislation.

Implemented Models

In most of the models we have developed so far, the OECD (1980) principles for fair information usage are taken as the top layer within the model; there is an intermediate layer of information analysis nodes and a lower layer of technological input. The lowest layer assesses information provided by agents, as described further below, including the configuration, presence and availability of PETs, and other IT controls, and the degree of evidence provided for security and privacy technologies. Part of such a model will be discussed in detail in a later section and is displayed, via our tool model editing interface, in Figure 4.

The lower level checks performed on the system will be composed of a mixture of checking for specific technology availability and its configuration. For example, corresponding to Figure 1 the specific low level checks we specified and implemented include the following different kinds of check:

1. The organisation has a privacy seal.
    (a) Presence of privacy seals for the back end system.
    (b) Validity of privacy seals for the back end system.
2. Organisation key service host computers are secure.
    (a) Key organisational services are patched up to the required level.
3. Organisation supports obligation management.
    (a) Organisation has support for obligation management via known adequate systems.
    (b) Systems version numbers are up to date.
    (c) Systems are available at least a given percentage of the time.
4. Organisation supports role based data access.
    (a) Organisation has support for role based data access via known adequate systems.
    (b) System version numbers are up to date.
    (c) Systems are available at least a given percentage of the time.
5. Organisation uses trusted hardware to enhance the security of key service hosts.
    (a) Organisation trusted hardware is of sufficient version and produced by a reputable company.
    (b) Organisation trusted hardware self test revealed no errors.
6. Key organisation services and resources operate in a safe environment.

(a) Key service hosts have trusted hardware installed.
(b) Key service hosts have adequate virus recognition in place.
(c) Key service hosts have properly configured firewall facilities.

Option to Focus on Specific Privacy Enhancing Technologies

Developing a comprehensive model is extremely complex, and so in many cases it makes sense to focus on specific privacy enhancing technologies and to be able to model the extent to which they contribute towards best practice, where the gaps remain, and whether they are deployed and operating correctly. For example, let us consider just two such technologies – privacy-enhanced access control (PAC), which puts a wrapper around "traditional" role-based access control, adding consideration of: the stated purpose to collect and hold personal information; data requestor's intent; data subject's consent details (Casassa Mont & Thyne, 2006) and a backend obligation management system (OMS), which provides a framework for monitoring, scheduling and enforcing responsibilities and duties related to personal information (Casassa Mont, 2005). The accountability provided by these technological controls with respect to the OECD principles would need to take account of the following:

- **Openness** Both PAC and OMS systems can contribute to this goal. Both systems provide a provisioning point for policy collection. As well as collection though, they also support the review and modification of those policies.
- **Use Limitation** PAC provides strong justification for use limitation as it allows the user to specify constraints on the usage of data by different roles. OMS also provides a weaker justification for use limitation as it can impose data retention restrictions on the organisation.
- **Individual Participation** The combination of PAC and OMS, as with openness, can provide justification for this goal as they provide participation from the user in the different areas of access and obligations.
- **Purpose Specification** PAC provides very strong justification for this goal as it defines a clear interface and enforcement of purpose specification.
- **Security Safeguards** Neither PAC or OMS provide any real justification for security safeguards. Other means must be used for the justification of this goal.
- **Accountability** PAC can support accountability by allowing an audit of the restrictions imposed by the data enforcer. PAC also supports logging of changes to policies.
- **Data Quality** OMS can provide some support for data quality through the use of notification of the user on data updates and active monitoring of obligation policies.
- **Collection Limitation** Neither PAC or OMS can be used to justify goals associated with this goal. Checking of collection limitation often requires an assessment process. There are possible ways that automated checks could be included but they could not, on their own provide comprehensive justification.

Model Specification

To make the analysis of the large set of fine grained information to be gathered from multiple heterogeneous sources, a comprehensive model needs to be developed. The model is flexible enough to encompass log or audit style information that is already generated by a number of products with a fine grained model of the configuration of resources. The modelling of resource configuration, in combination with the real time monitoring of the status of resources, allows a rich definition of the enterprise environment to be reasoned about.

In order to represent our assurance models we investigated two main types of approach: graphical notation and using the SmartFrog approach (an object oriented notation allowing representation of sets of attributes and values) (Hinde, 2005). In both cases there is an acyclic graph structure that links selected graph nodes, each of which has a type of information being modelled and a node-specific description of the desired state. This description can include properties such as controls, indicators, policies or the way system information should be processed. Information sources required for checking the state and the form of the output (resulting from matching the node description to the input) are defined for each node. The validity of the graph structure is checked using these inputs and outputs.

Our current approach to modelling is based upon a graphical modelling tool that has been developed. This tool allows the structure of the model to be created as a graph from low-level data source nodes being connected upwards to nodes representing high-level privacy concerns about the treatment of data. A Graphical User Interface (GUI) is used to permit different types of nodes and submodels to be placed in this graph. An initial set of node types allow various indicator statistics to be specified, along with policy combinations and control process definitions. Additional node types may be defined via extending the tool. The node inputs and outputs are described in terms of either a set of attribute value pairs or else columns, to give a better fit to a centralised enterprise database. The resultant model produced by the tool is in the form of an eXtensible Markup Language (XML) structure (XML, 2008) which can be input to the analysis engine, although it could be fairly easily modified to produce a model in the Smart Frog notation.

System Architecture

The architecture of our prototype system is shown in Figure 3. A full working prototype, based on this architecture, has been implemented.

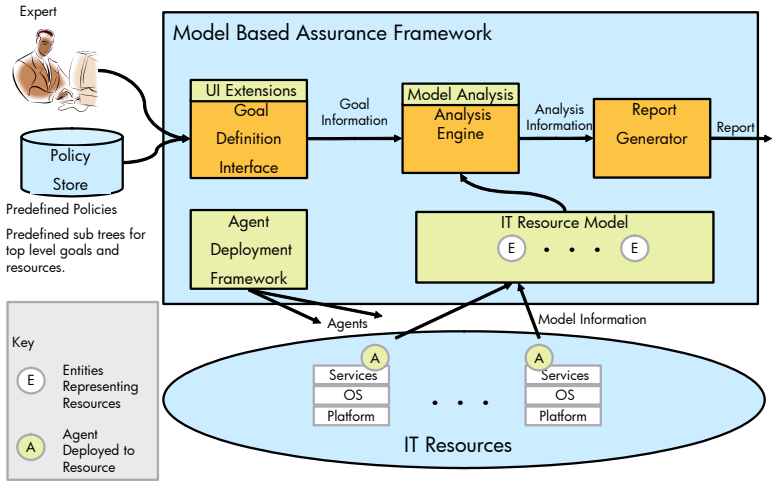

Figure 3. System architecture.

The main components of the architecture can be described as follows:

1. **UI Extensions** – This part of the system allows the construction of constraints over the set of resources present in the organization. This allows the modeling of privacy policies (satisfying requirement R1).
2. **Model Analysis** – This part allows the constraints defined in the goal definition interface to be applied to the model of the organization's key resources. The analysis engine populates the measured system with the relevant agents to collect the information needed, which is then collected at the analysis/backend of the system. The reasoning engine continuously monitors the state of the 'measured environment' against the states modeled. The top level goals are defined as a logical combination in order to specify the model and determine how to represent the effect in the dashboard. The analysis satisfies requirement R3 (analyzing relevant events).
3. **Agent Deployment Framework** – The agent framework is used to deploy software agents to the platforms' hosting resources that can monitor the events and configuration of the resources. The agents then map this information into the model of resources, thus helping satisfy requirement R3.
4. **IT Resource Model** – This provides a single model of the organization's IT resources. The model includes events captured from the resources, including from log files, and configuration information describing, for example, the setup of a host platform and runtime state. This is needed to satisfy R2 (the mapping of privacy models to IT infrastructure).
5. **Report Generator** – This generates reports, satisfying R4.

The architecture of the system is targeted at:

- Allowing the maximum flexibility of the resources that can be modeled. This is achieved by using agents to map information into a standardized form that can be processed more easily by the analysis part of the system.
- Avoiding the need to redevelop legacy systems to generate data – this can be achieved by using agents for data generation and sensing.
- Providing a single information format that is used to record information; this is mapped into by the agents or other sources.
- Providing an extensible framework for agent development.
- Generic analysis allowing the formation of complex queries over resource properties.

As shown in Figures 2 and 3, this system examines distributed system configurations using an agent infrastructure deployed across IT resources, feeds the findings into a reasoning engine and reports the resulting findings in a tree-like structure that can be 'drilled down' to the level of detail required. It uses functional decomposition to model privacy and model-based reasoning to carry out the analysis and generate reports. More specifically, modeling of privacy goals is combined with modeling of organizational resources and the processes around these resources. If desired, semantic web technology can be used to create a common understanding of lower level checks that are carried out, but this is unnecessary if the compliance checking is for internal organizational purposes only.

In our implementation the privacy models are represented in an XML format and the analysis involves Java and SQL queries. The modelling of organisational resources is achieved using a set of agents deployed through a RMI deployment framework to monitor resources in real time. The locality of entities is defined in terms of a Uniform Resource Identifier (URI) that defines the deployment location for an agent allocated to monitor an entity. Although our implementation used a relatively simple RMI framework, as mentioned above a more advanced agent framework like Smart Frog (Hinde, 2005), could be deployed: such automated monitoring frameworks allow monitoring of complex and dynamic systems.

In order to avoid the approach consuming too much time so that the running applications in the environment will be affected, the granularity of the measurement of the state of the environment can be tailored: for each case a 'reasonable' polling interval could be proposed or eventing system introduced. The model once defined in the backend analysis system is 'efficient' as it is expressed as a database query, which is run on changes registered to the environment.

Agent Infrastructure

Once the privacy expert has defined the model, the compliance checking system identifies the resource properties that have to be monitored for satisfaction of the policy constraints in the model. Given a model of how the policy will be satisfied, the system deploys a collection of software agents to monitor specific properties of the enterprise system. The agents also perform data normalisation of the PET and system properties prior to analysis.

Once an agent has been deployed to the platform on which it is to monitor a given type of entity it uses the platform and information passed to it when it was created to monitor the entity at that location. In this way, a host's agent is deployed to every instance of entities of the type host stored in the entity model. The agent, having been deployed, can then monitor the properties of the entity it is monitoring and update the model of the entity.

Agents deployed in this way to monitor organisational resources that we implemented include:

1. **Host agent.** This provides patch information and other information about the host entities described in the model, as input to analyse the integrity of the host systems which support key organisational services.
2. **PET status agent.** Each monitors the activity of a particular PET application (for example, its execution time).
3. **PET configuration agent.** Each monitors the configuration of a particular PET entity instance (for example, its presence, version number, etc.)
4. **Trusted Platform Module (TPM) agent.** This monitors the status and configuration of trusted hardware devices used in key resource hosts (for example, information associated with the vendor and status of a TPM).
5. **Certificate agent.** This monitors the organisation's certificates to assess whether they are current and valid.
6. **Privacy seal agent.** This monitors the status of the privacy seal certification to check that it is valid. The checking of the privacy seal validity may involve submitting a HTTP request to the Certification Authority (CA) to verify its integrity, and checking of the CA against a predefined list of CAs trusted by the user or administrator of the Compliance Checking System.

This approach is flexible, in that agents are relatively generic: if for example one was used for reading a specific log file, it could be used on other similar log files (with different recognition patterns). It would be possible to reuse agents as the set of developed agents became more extensive. The actual number of possible agent types is not limited.

SPCC Organization Representation Model

Following the convention of object representations of entities (i.e. specific resources), the database schema to hold the model of the organization's resources mirrors the object model used to represent the resources.

The database schema storing the model of the organizational resources uses the following schema:

```
Entity( entity_id, name, type, info, platform ) PRIMARY KEY enti-
ty_id;

Config( config_id, eid, presence, time ) PRIMARY KEY config_id
FOREIGN KEY entity_id REFERENCES Entity.entity_id;

Prop( prop_id, config_id, name, value ) PRIMARY KEY prop_id FOREIGN
key config_id REFERENCES Config.config_id;
```

```
StatusEvents( status_event_id, entity_id, status, time ) PRIMARY KEY
status_event_id FOREIGN KEY entity_id REFERENCES Entity.entity_id;
```

Where:

1. **Entity** The entity table holds information on each entity in the organization that is to be modeled.
2. **Config** The configuration table holds the configuration of entities described in the entity table. Configurations are stored for an entity and are relatively permanent. Example configuration details include operating system version number, virus recognition version number or TPM version/vendor.
3. **Prop** The prop table stores specific properties of configurations.

Using this association of entities with configurations and status-events allows the description of the properties of a specific resource to be arbitrarily large.

Interacting with the System

Figure 4 shows an example of the tool we developed to enable definition, input and customization of models that refine and transform privacy policies from high level statements to something that can be executed automatically at a lower level. In order to input the model, complete or partial graphs may be loaded. In addition, new nodes can be created via the buttons shown in the right hand side, each of which creates a different type of analysis node, and connections can be added via dragging and dropping arrow markers on the sides of the nodes. In order to configure these nodes, double-clicking on the nodes reveals windows appropriate to that type of node whose default settings can be changed. The models may be saved and/or exported for external usage.

Models can be created either in respect to a particular e-business process, or that incorporate multiple e-business processes in the same model, as desired. The model shown in Figure 4 focuses on assessing the deployment of a particular privacy policy enforcement system which is targeted at allowing both user preferences and enterprise policies to be taken into account when allowing access to personal information for a given purpose (blinded reference). We also developed other models, including analysis of a range of privacy and security-related IT controls and assurance information.

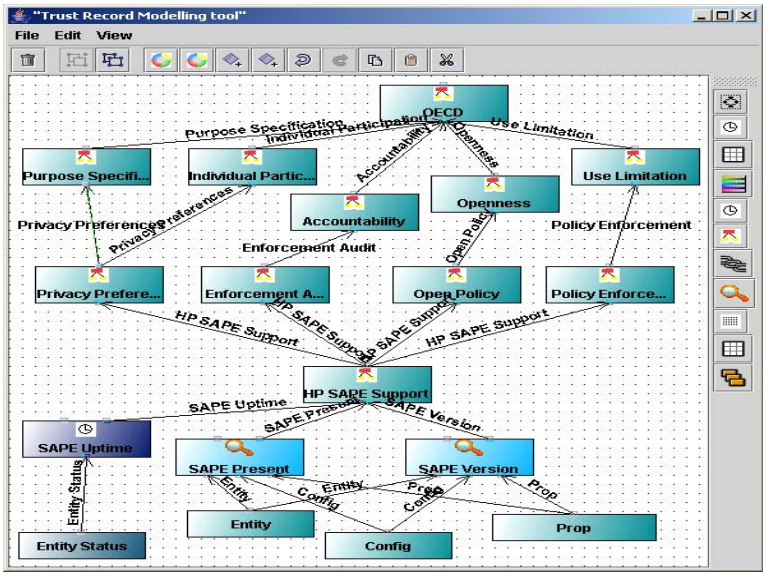

Figure 4. Example privacy graph.

Figure 4 shows various types of agent that produce data that can populate a node in the graphical submodel; other 'logical' nodes can be used to combine measured states or other derived type node (inputs could also be mixed). In the prototype a minimal set of logical operations were available (not, or, and, in, etc). The model for the particular graphical substructure shown in Figure 4 includes analysis of the extent to which a particular technology for privacy policy enforcement can contribute strongly towards satisfaction of use limitation and purpose specification and to a limited extent also to openness, individual participation and accountability. The justification for this is very similar to that already given in a previous section. There is no contribution to other OECD goals that form the top level of our extended privacy compliance model. Each of these goals is shown in a box in the top part of Figure 4. Below these, subgoals can be defined, also in boxes, that can combine logically to satisfy these higher level goals, and where this is the case a link will be shown connecting the subgoal to the higher level goal. There can be multiple subgoals, but in Figure 4 only the part of the model structure is shown that relates to Select Access Privacy Enforcement (SAPE), which is a product that provides privacy-enhanced access control (PAC) (Casassa Mont & Thyne, 2006). Lower level boxes in the model shown are of different types, in that they correspond to modeling at the IT level the presence, version and availability of the product. This figure is an illustration of the general procedure, as it shows just part of the more complex model that we used.

A key role is played by the privacy expert(s) who is in charge of creating models. This expert must have knowledge of privacy laws; understand relevant enterprise processes, solutions and systems; author models describing the expected behavior. It is unlikely that one person can have all this knowledge, especially in complex scena-

rios such as enterprises. More realistically we are looking at teams of people whose complementary knowledge can cover these aspects. In an enterprise context we believe that “auditing teams” satisfy these requirements. The model graph need not be constructed each time by experts, because it is possible to use ‘predefined’ parts of the system that can then be instantiated with configuration. Nevertheless, a new type of skill is needed to understand how to instrument properties of the model in the system.

Figure 5 shows an example of the compliance report generated by our system using the model shown in Figure 4. This report is targeted at company executives, managers and auditors in order to provide information in a transparent way that can highlight areas that are a privacy concern in a dynamic and accountable way, and allow drilling down if desired to obtain further levels of detail.

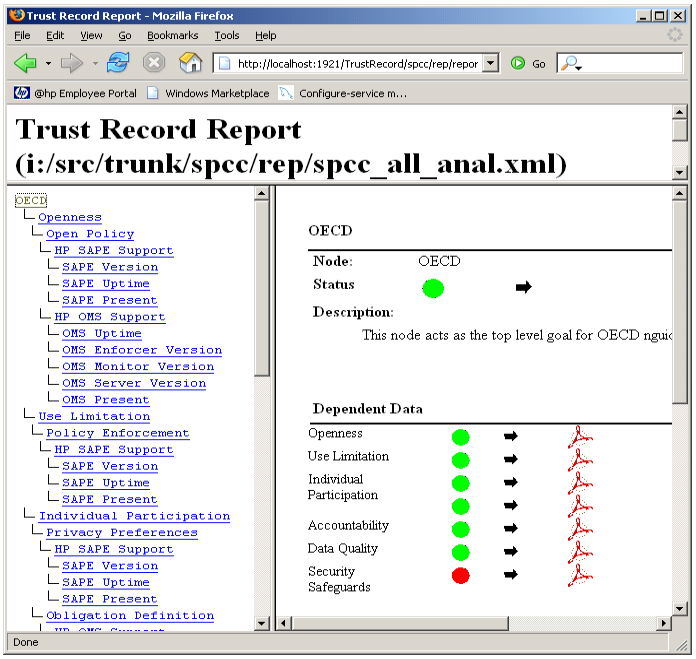

Figure 5. Example compliance report.

The report can indicate areas of concern, as well as showing the extent to which these problems affect higher-level privacy goals. In Figure 5, there is a security vulnerability but this particular problem is not judged to affect overall privacy compliance to a great extent.

Current Status

The Policy Compliance Checking System described in this paper is currently available as a prototype. There are a number of technical issues that would need to be addressed before this approach could be deployed, but the prototype presented does show our general methodology. The tool provides an (unchecked) high level reasoning framework that allows ‘experts’ to model the properties of a compliant system. Having the model means that it can at least be contested or ‘approved’ approaches may be suggested. The model then allows low level properties to be described and instantiated in the network. The model is in effect 'executed'. Changes in state are recorded in the backend. The model, when running is then evented on and changes reflected in the dashboard to register non-compliance. This does not have some of the properties that would be desirable in a productized version, for example a more reliable guaranteed agent communication mechanism, firewall traversal and encrypted communication using keys set up 'by hand' when the agent was installed and the system was considered to be in a secure initialization state. Instead, we focused our effort on trying to understand the usage of the interface to build our models. We are currently refining our modeling to reflect real-world scenarios, privacy regulation and best practice technologies.

For more detail about the prototype system design, see (Allison, 2005).

USAGE SCENARIOS

Our approach enables various usage cases that are centered around enterprise compliance with corporate governance legislation – such as Sarbanes-Oxley (SOX) and Health Insurance Portability and Accountability Act (HIPAA), enterprise policies and privacy legislation. It allows enterprises to determine whether system configurations or processes do actually conform to their assertions about privacy-respecting safeguards. This not only has an application in the auditing arena, but could also be used as a privacy expert system highlighting areas for improvement. Furthermore, there are certification service opportunities for development of the model and component substructures, agents and analysed systems, and opportunities for semi-automation of the provision of privacy seal or best practice certificates.

In addition, our solution could be adapted to increase user trust and willingness to engage in e-commerce. For example:

- Giving consumers the ability to determine whether unknown vendors on the Web are using IT systems and processes that can be trusted to execute their stated privacy policies.
- Automation of privacy assessment of the service side can be conveyed to the user in an open way (i.e. the compliance reports can be accessible to public) and with much more of a focus on evidence rather than having to rely on self-certification.

Overall, there can be seen to be two main benefits of our approach for enterprises. The first relates to the formalization of privacy models. Currently, documentation of compliance approval processes (especially privacy seals) is a manual process which is

aided by intuition and tacit knowledge. Even without the monitoring aspects of our solution, this process of decomposing privacy satisfaction is extremely powerful and can have value for business, for example in allowing links to be made across different audits or in aiding consolidation of information during a merger.

There is additional business value using an approach like ours because it allows justifications to be audited rather than just the IT infrastructure, and furthermore allows assessment to be much more continuous and responsive to change. Combining this huge increase in functionality with automated analysis can save vast amounts of time and monitoring effort for large enterprises.

RELATED WORK

Our goal is to provide an automated policy compliance checking system that can include checks about trust and assurance properties and that takes into account changing information, including at the IT resource level. Our compliance checker is based upon a model-based assurance framework that provides generic assurance modeling and analysis; furthermore, it is focused on privacy and can model organizational resources and reason about system and application properties. We are not aware of products or solutions providing this type of model-driven assurance and compliance verification.

There has been a great deal of work done on defining privacy polices: policy specification, modeling and verification tools include EPAL (Ashley et al, 2003), OASIS XACML (2005), W3C P3P (Cranor, 2002), Datalog with constraints (Li & Mitchell, 2003) and Ponder (Damianou *et al*, 2001). In these policy frameworks the focus has been on access control based on conditional logic. The high-level goals associated with the upper levels of nodes in our models are not this kind of policy, and it is not appropriate to process them against some rule set to produce a decision on whether data should be released. Rather, they correspond to goals within a control framework: descriptions of desired privacy features, corresponding IT controls and possible implementations and configurations of these. The model itself would be the subject for agreement by a privacy or audit expert as to its validity, rather than something which is in itself automatically provable. The approach presented here is intended as an aid to locating gaps in compliance and potentially also to highlighting operational privacy problems; it would not be comprehensive, or fully automatable, or suitable to be the subject of formal method-type proofs. What the model does do is to drive the subsequent analysis and reporting, as described above.

P3P specifications (Cranor, 2002) allow people to describe their privacy expectations and match them against the level of privacy supported by an enterprise. This helps shape people's trust in enterprises. However, P3P only checks if their expectations are matched against promises made by the enterprise, and does not provide mechanisms to check and prove upfront compliance with fine-grained constraints. As is the case with privacy seals, P3P cannot link the privacy practices expressed by the website and anything tangible on the back-end. Our solution can actually be used to fill this gap, in that it is capable of providing assurances that are missing from the P3P model. That is to say, as already discussed in the usage scenarios section, the system

described can be used internally within an organization to assess internal compliance, but could potentially as well be used to advertise that compliance to other parties as part of an assurance provision process. For example, it can be used to enhance the compliance checking system aimed at giving users control over compliance checking of organisations that they wish to interact with that is implemented as a subpart of the PRIME system (PRIME, 2008).

This approach requires privacy experts to input the models by hand. There is therefore scope to improve this process by including more automation at this stage. Annie Anton and co-workers have spearheaded research into how to automatically extract rules and regulations from existing natural language text (Breaux & Anton, 2008), although it would require further research to assess how exactly that work might be used in the current context. Other related work is on policies and iconography, notably that of Mary Rundly to learn from creative commons licences and use icons to express different policies (Rundly, 2006).

Other compliance checking products hardcode their compliance checking process or at least cannot model privacy processes and IT components as we do. They are targeted at the definition and monitoring of compliance goals for all IT related organizational resources, whereas our system focuses on monitoring the privacy compliance of key resources. More specifically:

- *Computer Associates Policy and Configuration Manager* (Computer Associates, 2008): this offers centralised security policy and configuration management, but does not seem to provide the infrastructural support for auditing of regulatory compliance. It can identify resources that have not been correctly configured. It focuses on developing a security enhancing solution rather than a governance tool.
- *HP OpenView Compliance Manager* (Enterprise Management Associates, 2005): this is a report pack based on OpenView that provides metrics for internal audit at the infrastructure level.
- *IBM Security Compliance Manager* (IBM, 2008): early warning systems identify security vulnerabilities and security policy violations and support compliance definition and monitoring. This integrates with Tivoli's automated security management tools to help mediate security policy violation and risk and uses predefined policies based on SANS top risks to security and compliance.
- *SenSage Compliance Solution* (SenSage, 2008): this system uses event log data for analytics, and provides reports for audit.
- *Sun Java System Identity Auditor* (Sun Microsystems, Inc., 2005): this aims to help compliance with internal and external regulatory requirements across critical enterprise applications and across the identity management infrastructure. It features a compliance dashboard, an audit scan and reports.
- *Synomos Compliance & Data Governance* (Synomos, 2006): this was a system for managing data policies and compliance. It was policy driven rather than model driven and was focussed on getting low level events. It cannot model processes or check with the level of granularity of our system. This system is no longer on the market.

Steps towards the provision of more assurance to people on privacy have been made by various privacy seals providers and verifiers (Cavoukian & Crompton, 2000). This approach provides general purpose information about the conformance of a service provider or an enterprise with certified, privacy compliant processes when handling and managing personal data. However, the information is nearly always produced by self-certification and cannot be checked dynamically.

In summary, the differentiating features of our technology are the following:

- it is privacy-focused
- it is model-based and uses functional decomposition of privacy goals and constraints
- it allows checking of trust and assurance constraints
- it allows the combination of runtime state, process analysis, log data, resource and infrastructural models and other information sources into a explicit representation of how an enterprise satisfies its obligations
- it neither presupposes deployment of other proprietal products nor requires major changes to applications, services or data repositories
- it can provide stronger degrees of evidence
- it can provide a finer level of granularity
- it is not reliant on self-certification and people-driven processes

LIMITATIONS TO AUTOMATED REASONING FOR PRIVACY COMPLIANCE

In the course of this research, various constraints became apparent that limit the effectiveness of automating privacy compliance assessment:

First of all, we are able to provide partial automation only because of a lack of formal verifiable definitions of manual processes that are currently used to check the validity for example of privacy seals, and these are difficult to automate, and also because manual process entries are sometimes necessary, and in these cases the most we can do is to have a website automatically generated to request such information.

Second, there is complexity involved in the necessary modelling, and so it can be difficult. For example, back end infrastructures can be extremely complex: to reduce this problem we model just the key privacy-related subparts of such systems. It is also necessary to address the complexity of how subnodes within the privacy models relate to each other.

Third, there is some missing infrastructure currently. There is a need to standardise a meta-data format for machine-readable certificates because most machine-readable certificate information that is available is not very interesting from a privacy point of view, and other interesting information is not machine-readable or analysable.

In addition, there is no trusted infrastructure around agent deployment, and so the information obtained from the agents cannot be trusted. The problem is that malicious layered services could operate unknown to (our) monitoring services — there is a risk of administrators compromising the system and also a risk when checking external

topologies. This problem is generally faced by compliance monitoring and reporting systems. Approaches to solve this include:

– Authentication between components (which may be enhanced by using trusted hardware to protect private keys)
– Next generation trusted computing and infrastructure, e.g. Trusted Computing Group (2003) integrity checking (if available, allowing the verification of a loaded system image to avoid system compromise), agents isolated in trusted compartments (Anderson, Moffie and Dalton, 2007), etc.

Further Development of our Research

Given these issues, we see potential usage of this technology as developing over time in the following way:

- In the *immediate term*, basic system properties, such as presence, availability and properties of services, security hardware, etc., could be checked, together with configuration (e.g., patching). Also monitoring of changes to the infrastructure over time and assessment of logs. Where privacy enhancing technologies are not available due to lack of deployment or even availability in the marketplace, the equivalent manual processes can be included and monitored within the submodels.
- In the *intermediate term*, schema definitions could be provided for properties of IT controls, improve risk assessment models for privacy and audit what really happens against expected enforcement (for example, by checking failures in a specific PET).
- In the *longer term*, more PETs and technologies could be assessed as they reach the market (so that both these technologies and alternative human-driven processes for the same business processes are modeled), data flow could be modeled and instrumented and a trusted infrastructure could be used to protect the agents.

The technology would be particularly suitable for areas such as enhancing privacy for customer relationship management and enabling checks of compliance by partners with whom data is shared. It could also be applied in a focused way to situations such as identifying issues before an upcoming audit or before problems arise, trying to prevent fraud and internal threats and checking that important data (such as forensic data) is kept in a suitably protected way.

CONCLUSIONS

This project has succeeded in demonstrating the feasibility of a model based privacy 'best practice' compliance checker by extending what is automatable. This was achieved by encoding a set of high level goals, based on guidelines defined by the OECD (1980) and linking these to system level enforcement technologies that may be used to satisfy these top level goals. We proposed a framework that helps experts reason about how privacy compliance may be satisfied, optionally using prede-

fined/approved submodels. Once defined, the resultant model can then be executed and generates a 'compliance dashboard' that can be used by non-expert users.

Working prototypes have been fully implemented to demonstrate the feasibility of our approach. Even subparts of such a system prove useful.

The use of an agent framework has shown that it is possible to instrument the collection of fine grained data regarding organisational resources. This in combination with simulated data has allowed the demonstration of the failure of entities to pass constraints. The use of the reporting system has demonstrated how the information regarding failures can be reflected at a high level, whilst allowing the user of the system to explore the specific cause.

The development of a framework for the formal definition of privacy and security constraints forces the explicit description of satisfaction constraints. The use of such a system as part of a risk analysis and mitigation framework can be of significant benefit. The automation of analysis and reporting would provide a more specific measurable assessment of an organisation's compliance.

REFERENCES

Allison, D. (2005). *System Policy Compliance Checker* MSc.Thesis, University of Newcastle, UK.

Anderson, M.J., Moffie, M., and Dalton, C.I. (2007). *Towards Trustworthy Virtualisation Environments: Xen Library OS Security Service Infrastructure* (Tech. Rep. No. HPL-2007-69). HP Labs, Bristol. Retrieved June 2008 from http://www.hpl.hp.com/techreports/2007/HPL-2007-69.pdf

Ashley, P., Hada, S., Karjoth, G., Powers, C. & Schunter, M. (2003). *Enterprise Privacy Authorization Language (EPAL 1.1)* (Research Report). IBM. Retrieved Jan 2008 from http://www.zurich.ibm.com/security/enterprise-privacy/epal

Breaux, T.D. & Anton, A.I. (2008). Analyzing Regulatory Rules for Privacy and Security Requirements. *IEEE Transactions on Software Engineering*, 34(1), 5-20.

Casassa Mont, *M* (2005). *Handling privacy obligations in enterprises: important aspects and technical approaches*. Comput. Syst. Sci. Eng. 20(6).

Casassa Mont, M., Thyne, R. (2006) *A Systemic Approach to Automate Privacy Policy Enforcement in Enterprises*. Privacy Enhancing Technologies 2006, LNCS 4258, eds. G. Danezis and P. Golle, Springer, 118-134.

Cavoukian, A., & Crompton, M. (2000). Web Seals: A review of Online Privacy Programs. In *22nd International Conference on Privacy and Data Protection*. Retrieved Dec. 2006 from http://www.privacy.gov.au/publications/seals.pdf

Computer Associates (2008). Policy and Configuration Manager. Retrieved June 2008 from http://www3.ca.com/solutions/Product.aspx?ID=165

Cranor, L.F. (2002). *Web Privacy with P3P*. O'Reilly and Associates.

Damianou, N., Dulay, N., Lupu, E. & Sloman, M. (2001). *The Ponder Policy Specification Language*. Retrieved 2007 from http://www-dse.doc.ic.ac.uk/research/policies/index.shtml

Enterprise Management Associates (2005). *HP Openview Compliance Manager : Integrating the Synergies of Management, Security and Compliance*. Retrieved June 2008 from http://www.managementsoftware.hp.com/products/ovcm/swp/ovcm_swp.pdf

European Parliament and Council (1995). Directive 95/46/EC on the protection of individuals with regard to the processing of personal data and on the free movement of such data. *Official Journal of the European Communities.* L. 28, (p. 31).

Greenleaf, G. (2005). APEC's Privacy Framework: A new low standard. *Privacy Law and Policy Reporter*, 11(5). Retrieved 2008 from http://www.aph.gov.au/Senate/committee/legcon_ctte/completed_inquiries/2004-07/privacy/submissions/sub32ann_c.pdf

Hinde, S. (2005). *The SmartFrog Reference Manual*, v3.06. Retrieved Dec. 2006 from http://cvs.sourceforge.net/viewcvs.py/*checkout*/smartfrog/core/smartfrog/docs/sfReference.pdf

IBM (2008). *Tivoli Security Compliance Manager*. Retrieved June 2008 from http://www-306.ibm.com/software/tivoli/products/security-compliance-mgr/

Information Commissioner's Office (2007). *PIA handbook*, UK. Retrieved November 2008 from http://www.ico.gov.uk/.

Laurant, C. (2003). *Privacy and Human Rights 2003: an International Survey of Privacy Laws and Developments*. Electronic Privacy Information Center (EPIC), Privacy International. Retrieved Dec. 2006 from http://www.privacyinternational.org/survey/phr2003/

Li, N., & Mitchell, J.C. (2003). Datalog with Constraints: A foundation for trust management languages. In *Proc. PADL'03* (pp.58-73). Springer Verlag.

OASIS (2005). eXtensible Access Control Markup Language (XACML). Version 2.0. Retrieved Feb. 2005 from http://docs.oasis-open.org/xacml/2.0/access_control-xacml-2.0-core-specs-os.pdf

OECD (1980). *OECD Guidelines on the Protection of Privacy and Transborder Flows of Personal Data*. Retrieved Dec. 2006 from http://www1.oecd.org/publications/e-book/9302011E.PDF (1980)

PRIME (2008). Privacy and Identity Management for Europe. European RTD Integrated Project under the FP6/IST Programme. Retrieved June 2008 from http://www.prime-project.eu.org/

Rundly, M. (2006). International Personal Data Protections and Digital Identity Management Tools. In *Proc. W3C Workshop on Languages for Privacy Policy Negotiation and Semantics-Driven Enforcement*. Retrieved Jan. 2008 from http://www.w3.org/2006/07/privacy-ws/papers/21-rundle-data-protection-and-idm-tools/

SenSage (2008). SenSage 4.0 Product. Retrieved June 2008 from http://www.sensage.com/

Sun Microsystems, Inc. (2005). *Identity Auditing : Taking Compliance Beyond the Baseline*. White Paper. Retrieved Dec. 2006 from http://www.sun.com/software/products/identity_auditor/index.xml

Synomos (2006) Synomos Align 3.0, Retrieved Dec. 2006 from http://www.synomos.com/

Trusted Computing Group (2003). *TCG Main Specification*, v1.1b. Retrieved Dec. 2006, from http://www.trustedcomputinggroup.org

Wikipedia (2008). Electronic business. Retrieved June 2008 from http://en.wikipedia.org/wiki/E-business

XML (2008). Extensible Markup Language. Retrieved June 2008 from http://www.w3.org/XML/