electronic Journal of Health Informatics
http://www.ejhi.net
2008; Vol 3(1): e4

Using a Client-Task Based Approach to Achieve a Privacy Compliant Access Control System

Leigh de la Motte, Jacky Hartnett
*School of Computing, University of Tasmania,
Launceston, Australia*

Abstract
This article proposes a solution to the problem of assuring the privacy of low value client information such as that maintained by a hospital. The proposed solution involves the development of a compliant low-cost system. It is based on the fundamental requirement that such a system needs to provide integration, generalization and inbuilt consent. ***Integration*** *brings together the technical, managerial and regulatory components of an organisation's system.* ***Generalization*** *provides all the access control functionalities that are necessary for the system to be useful in a diverse range of organisations.* ***Inbuilt consent*** *ensures that data owners consent to the use of their personally identified data. The* ***Integrated System*** *proposed here uses a* ***Client-Task*** *approach. It is based on the observation that a client is not a user of the system yet has a form of ownership over their personally identified data held within the system. Furthermore, in industries such as health, professionals and managers often determine who has access rather than systems administrators.*

Keywords: Authorization, tasks, groups, context, roles

1. Introduction

The increase in personally identified client data managed by computer systems has meant the cost of providing data security has become a significant issue to organisations. One of the purposes of this paper is to outline the problems that must be overcome in order to provide cost effective privacy for client data. The paper deals with the protection of electronic records which is a significant issue in contemporary healthcare. Our approach is to take a broad look at the problem and to incorporate diverse aspects of access control into a general solution. This is in contrast to methods which tailor solutions to specific problems.

In this paper "privacy" should be interpreted as a legal term while "security" should be interpreted as an IT concept relating to access control. This means that in terms of enabling the desired privacy goals within computer systems, system security should be viewed as the means to privacy protection. The paper also seeks a flexible solution where a range of regulations or legal rulings can be modelled in the system rather than adopting just one particular approach. This enables the system to cope with regulatory changes.

There are two aspects of security to consider. The first is to restrict access to the system to legitimate users. This problem is handled by employing appropriate authentication techniques. The second is to further restrict access to legitimate users on a need-to-know basis. This problem is handled by authorization techniques. Our research deals with the second problem of authorization. Our primary motivation is to provide fine-grained access to health records based on a worker's role and their membership of a treating team.

When we look at the cost of implementing system security, Lampson [27] states that:

> *"Practical security balances the cost of protection and the risk of loss, which is the cost of recovering from a loss times its probability.....When the risk is less than the cost of recovering, it's better to accept it as a cost of doing business..... than to pay for better security."*

This means in practical terms that the lower the value of the information that is to be protected, the lower the cost of administration must be. Legislative requirements that many countries have adopted, based on the OECD data security principles [32], have provided an impetus for organisations to try to improve their security

The *electronic Journal of Health Informatics* is an international journal committed to scholarly excellence and dedicated to the advancement of Health Informatics and information technology in healthcare. ISSN: 1446-4381

protection. However, in many practical cases where the security requirements are complex (fine-grained), the cost of administration can only be reduced to suitable levels by compromising security. Consider a situation where all clinicians in a hospital have access to the complete medical records of all of the patients in the hospital, regardless of their roles and whether or not they are treating them.

The basic problem then is, "How can the privacy of low value information be assured?" This problem is particularly relevant to the Health domain that we are concerned with, where personally identified patient data is generally of low financial value. This is not to say that such information is not important, but that in the vast majority of cases no-one would be prepared to pay much for the information. From the view of medical practitioners and managers the problem can be seen as, "How can protection of digital data be provided for reasonable cost?"

There are two fundamental techniques which can be applied to solve the "low-value protection problem". The first technique is to increase the "apparent value" of the information by instituting monetary penalties for privacy breaches and/or non-compliant systems. This paper deals with the second technique, the development of a compliant low-cost system. The system proposed employs a "Client-Task" approach to access control. It seeks to address the two fundamental issues of compliance and cost.

1.1. Compliance

The compliancy aim and basic methodology used here are similar to that of "the formal task-based privacy-model" proposed by Fischer-Hübner and Ott [22]. However, our approach focuses more on consent. It essentially revolves around the concept that:

> *Privacy compliance fundamentally requires data owners to consent to the use of their personally identified data.*

This concept effectively describes what access control is all about. "Compliance" implies that there must be an appropriate legislative & policy framework in place. In order to define if an access is compliant, data ownership within a system must be accurately specified. In this context data ownership refers to the name specified as the owner of the data/file within the system rather than to any legal type of ownership, though the two may be related. Our view is that the individual described in the data, the client, should be given some form of ownership of their data within the system. Finally, the concept states that consent from the owner is required for all accesses to personally identified data. This means that consent must be seen as part of the system. This leads our first requirement:

- *Requirement 1:Compliance requires inbuilt consent.*

Data ownership is a complex issue and is beyond the scope of this paper. We assume client ownership of data here as this type of ownership is not dealt with, to our knowledge, in current models.

1.2. Low-Cost

There are two types of system costs – "setup cost" and running costs ("overheads"). This leads to another requirement:

- *Requirement 2:Low-cost requires low setup cost and low overheads.*

The setup cost of a software system depends on its complexity and on its reusability. A low cost system must be as simple as practicable, and usable in many different environments. A simple access control solution which can be applied generally to organizations is needed. This gives us our next requirement:

- *Requirement 3:Low setup requires generalisation.*

In order to have low overheads in a complex system manual tasks must be efficiently automated. The process of automation can only work efficiently if all parts of the system are able to interact effectively with other parts of the system. For example, service-related operations need to trigger other related operations such as the attainment of consent. This leads to the requirement:

- *Requirement 4:Low overheads require integration.*

1.3. A Compliant Low-Cost System

Reiterating, the goal of this work is the development of a compliant low-cost system. The previous requirements lead to the principle requirement:

- Requir*ement 5:Low-cost compliance requires an integrated general* system with inbuilt consent.

This is a conceptual paper. We are deliberately looking at the whole problem with a broad view using a "top-down" approach. Our solution essentially revolves around client (patient) consent being given in terms of the approval of a worker's membership of a client service team (treating team). Further access restrictions are then based on the worker's role or other group memberships. The cost problem is addressed through enabling managerial and professional control of group memberships, much of which can be automated through triggering by workflow tasks.

In the next three sections we look at the three components of

- *Requirement 5 – integration, generalization and inbuilt consent.*
  These are the essential requirements needed to achieve compliance at low-cost.

2. Integration

Any type of integration of parts requires that commonalities between the parts are found and used as the basis for integration. It is necessary to examine each part in order to tease out the commonalities. Since we are working conceptually we want to consider the nature of the parts in our systems rather than specific details of how they work. We therefore subdivide the parts of the system on the basis of their fundamental purpose. We introduce three "views of access control" to describe purpose. The three views are described in the following sub-section. Once they have been described we will discuss how they can be integrated.

2.1. Views of Access Control

2.1.1. Technical View

The technical view relates to how systems administrators deal with access control. The vast majority of current access control systems use the Subject-Object approach where users (subjects) are granted privileges (rights) to access information (objects).

System Administrators are charged with the task of managing the organisation's system. They have an inherent responsibility to provide protection for the information contained within the system. The continual information demands of users often weigh against this need for protection. This can result in the balance between protection and availability being tipped in one direction. Imbalance produces systems where it is difficult for users to gain access to the information they need or systems where protection of information is limited.

Most systems put the onus on setting up access rules in advance. There is often little or no scope for dealing with unpredictable access requirements. Figure 1 shows the components of a simple access control system. Administrators are responsible for storing and managing the files (objects), System Administrators are responsible for granting privileges to users and for managing system auditing, and Staff Managers are responsible for allocating jobs to users and dealing with audit results. It may be that in particular organisations the System Administrator(s) perform two of the three or even all of these roles.

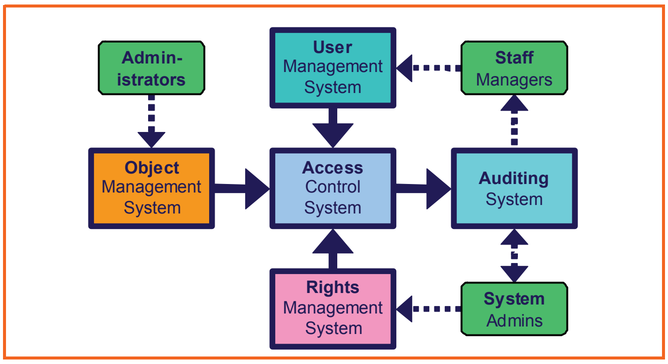

Figure 1: Technical View

2.1.2. Management View

Managers are the people in the organization who are essentially responsible for maintaining productivity and making spending decisions. There is naturally a tendency for them to concentrate on service-related tasks. They are interested in job allocation & completion. System security can be seen as imposing unnecessary work and cost.

Figure 2 shows the components of a simple management system. It illustrates how the roles of staff managers are described in the Organisation Management Structure and that they are primarily responsible for the allocation of clients and tasks to workers. The diagram also shows the links between the management system and the Access Control and Consent Management Systems.

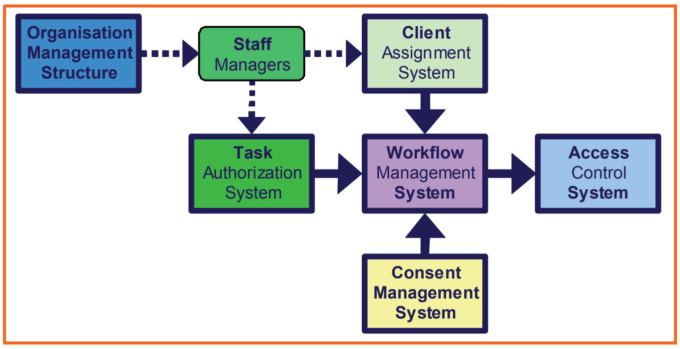
Organisation Management Structure
Staff Managers
Client Assignment System
Task Authorization System
Workflow Management System
Access Control System
Consent Management System

Figure 2: Management View

2.1.3. Regulatory View

Governments and professional associations are increasingly introducing regulations that impose conditions on how workers may access personalized data. The basic view is that private information must be protected.

Regulations are put into place to enforce the Principle of Least Privilege which requires that accesses be allowed on a need-to-know basis. They may specify the conditions where consent must be obtained and outline what constitutes consent. They seek to ensure that data owners are informed of who can see their data and what they can do to restrict access to sensitive information. Regulations limiting the secondary use of data are also common.

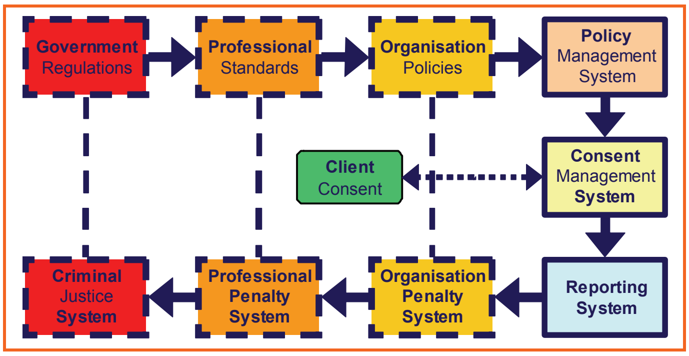

Figure 3: Regulatory View

Figure 3 gives an outline of the regulatory view. The three dashed rectangles at the top show three levels of rules that impinge upon system design and use. The three dashed boxes at the bottom show the associated penalty regimes which apply when the rules are breached. The boxes on the right show system components which enforce the required rules and report any rule breaches.

2.2. An Integrated System

2.2.1. Joint View

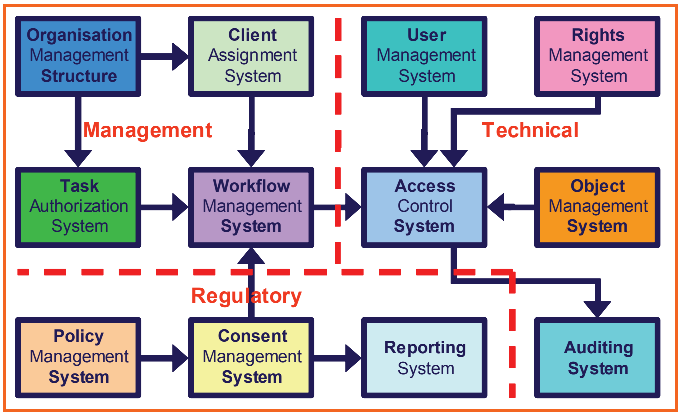

Figure 4: Joint View

The Joint View shown in Figure 4 is a compilation of the Technical View of Figure 1, the Management View of Figure 2, and the Regulatory View of Figure 3. It shows the boundaries and links between the three views.

In most organizations the components of the system vary between being paper-based and computer-based. There are also a variety of paper-based and computer-based systems. In general terms the Technical components are “rights-based”, the Management components are “task-based”, and the Regulatory components are “rule-based”. The fundamental problem to solve is how to incorporate all these differing components in a way that they can interrelate. The key is to find “unifying abstractions” which are relevant to all three sections of the Joint View.

2.2.2. The Common Vocabulary

The "common vocabulary" introduced here seeks to use terms that are meaningful to regulators, managers, system administrators, workers and clients. It seeks to describe the abstractions that unify the technical (access control), managerial (workflow) and regulatory (compliance) aspects of the system, allowing diverse system components to work together.

2.2.3. The Client-Task Approach

Figure 5 describes the Client-Task approach. The following statement defines the approach. The words in italics that follow form the basis of the common vocabulary:

Managers (supervisors/administrators) employed by the *organization* authorize *workers* (employees/contractors/ users) to perform (service-related) *tasks* for the organisation's *clients* subject to the organisation's *policy* which complies with appropriate *regulation*.
Workers and Managers can be placed in *groups*. Workers and managers perform in their *roles* (positions) within the organization. Roles can be represented as a type of group. Each group is authorized to perform a set of duties (tasks). Specific duties (*client-tasks*) are assigned to individual workers.

The concept of using tasks for access control is not new (see [22, 41]), but the concept of incorporating clients who are data owners who have no direct access to the system may be. In the workplace a task can be thought of as a particular "duty". It often has a manual component and an IT component. For example, if a nurse has the duty of giving a patient an injection, the nurse is the Worker, the patient is the Client and the giving of the injection is the Task. The manual component of the task is the physical administration of the injection while the IT component may be some sort of record of the task. In terms of the computer system we are only interested in the IT component of the task. The task in Figure 5, *Task A*, therefore refers to the IT component of the task.

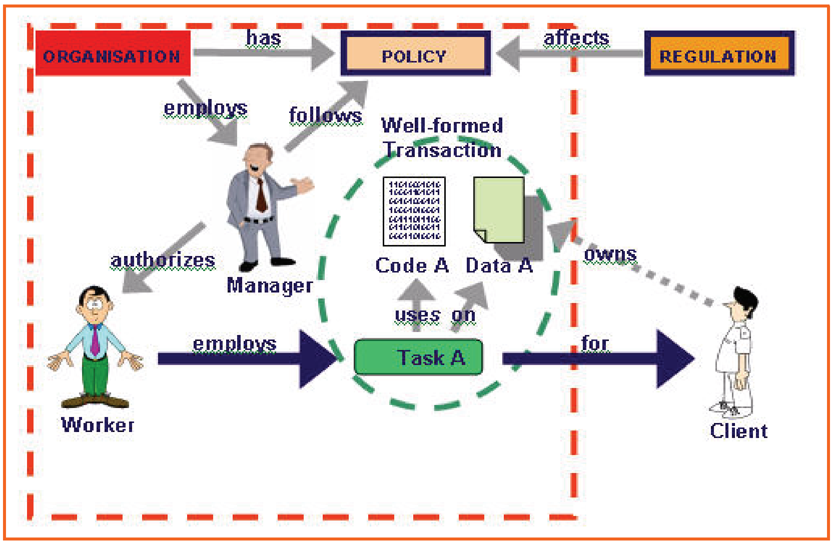

Figure 5: The Client-Task Approach

Task A is essentially a process which utilises *Code A* to manipulate *Data A*. The task is analogous to a "well-formed transaction" as defined in the Clarke-Wilson model [10], because the worker employs a task to manipulate the data and only has access to the data through the task.

The traditional Subject-Object view of access control says that "a subject can be assigned a right to access an object." The "Client-Task" view of access control says that "a worker can be assigned a task to perform for a client." *Task A* can be thought of as having a *right* to access *Data A*. In Subject-Object terms the *Worker* is the Subject, *Task A* is/has the access right and *Data A* is the Object. The *Client* is the *owner* of *Data A* and is outside the *Organisation* (represented by the dashed red square in Figure 5).

2.2.4. The Integrated View

Now that we have defined the methodology of an integrated system, we can describe the components of such a system. The Integrated View of Figure 6 shows how the Joint View of Figure 4 can be adjusted so as to facilitate the integration of all the required system components. The principal change is the combining of the Workflow Management and Access Control components into a Central Control System.

In another change, the User/Worker Management System is relocated to a place where it is more a management component rather than a technical component. This reflects our view that Organisational Managers should control "Worker Management (placing workers in groups)" whereas System Administrators should control "Rights Management (assigning tasks to groups)".

The Task Creation & Allocation component is so-named to reflect the idea that the system is based on the Client-Task approach. A new connection between the Auditing and Reporting components is made because there is no need to duplicate reporting facilities in the Auditing component.

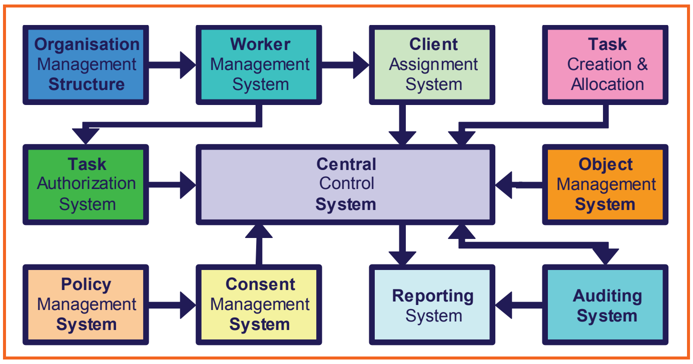

Figure 6: Integrated View

3. Generalisation

Our research into current access control models has revealed that in order to achieve fine grained access control in all situations, there are fundamental criteria that must be met. The general solution that we propose is based on the criteria derived from existing models. The problem appears to us to be that none of the existing models have sought to incorporate all these criteria. This is the fundamental reason why they cannot efficiently control access to personally identified data based on service team membership and worker role in all required circumstances.

An efficient solution requires generalisation to facilitate the automation required to cut costs to acceptable levels. Rather than tailoring a solution to meet the needs of a particular hospital, it would be more cost effective to have a solution which meets the needs of many hospitals. It would be more cost effective again to have a solution which meets the needs of a diverse range of organisations. Generalisation seeks to find such a broadly applicable solution.

This section outlines the criteria for a general purpose access control solution which satisfies our treating team scenario. Each sub-section details a required criterion and shows the current models which address that criterion.

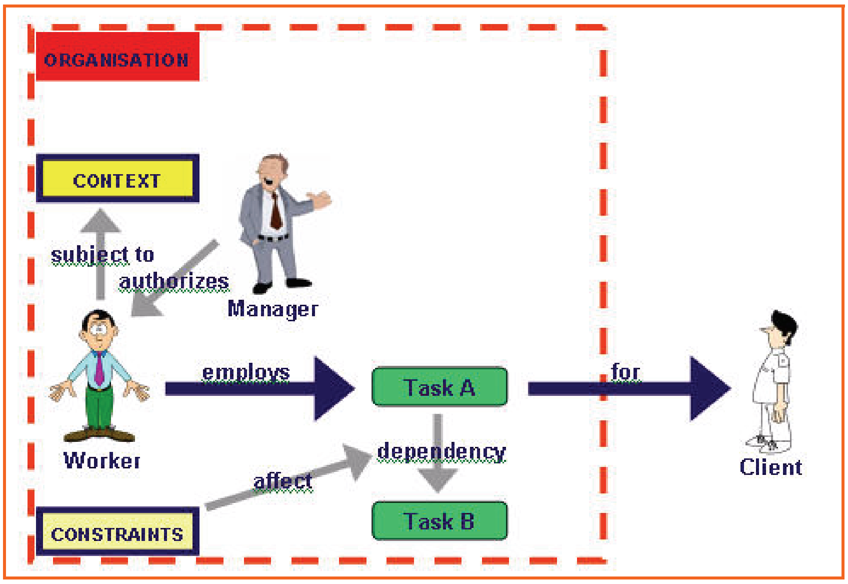

Figure 7: Context and Constraints

3.1. Multiple Access Control Contexts

Figure 7 shows that authorizations are given subject to the context. Contexts essentially represent some attribute that the worker has. Contexts are independent in nature in that one context does not affect another. For example, the Worker's role does not affect their location, though both may be relevant to determine if an authorization is to be given. It is our view that all contexts can be represented by group memberships. This will be investigated in ongoing work.

Many RBAC based models have used context constraints to model complex access control needs. Examples are [23], [4], [5], and [31]. The basic idea is that as well as using role-based controls, certain context constraints are checked at runtime to determine whether accesses should be permitted.

In order for fine-grained control to be achieved in an access control system it needs to be possible for the system to take more than one context into account. For example, an access request may need to take the worker's role and their location into account before granting access. It must be possible to form complex rules. The problem is that complex rules normally incur inordinate administrative overheads. Simple administrative techniques which have a degree of automation are required to ensure that overheads are acceptable.

A further issue here is how to store the rules and how to process access requests. This is complicated when inter-domain access is also required. The question is which domain's rules are used and how do rules take other domains into account.

Attribute Based Access Control [42] looks at the inter-domain problem. Rule-Based RBAC approaches [26] [1] [16] are also of interest, as is Organisation Based Access Control (ORBAC) [17].

3.2. Interdependence of Tasks

While tasks are analogous to well-formed transactions, they have an important additional property – they can be dependent upon one-another. This is a vital property which can be used to model access control constraints such as the Separation of Duties. Figure 7 shows the dependencies between different tasks.

Examples of the types of task dependencies are:

- Mutual exclusion between tasks;
- Prerequisite tasks;
- Tasks triggering other tasks;
- Tasks having Sub-tasks;
- Limits to the number of tasks performed; and
- Task allocation methods.

Work in the areas of Workflow Management Systems (WMSs) [37] [6] [7] [3] [25] and Access Control Constraints [4] [5] [31] [9] have relevance to this discussion.

3.3. Inter-Domain Access

Inter-domain access allows for information to be accessed from outside a domain. Networks in many organisations are now accessed from outside the organisation.

It is common for many copies of information to exist in diverse places. When the original information is updated at the source the copies do not reflect these changes. A system which allows inter-domain access can mean that the copies can be automatically updated. Alternatively, the source can be accessed when the information is needed meaning that copying is not needed.

There can also be a need to perform tasks remotely. For example, it is common for doctors to authorize procedures remotely.

The concept of Web Services also requires inter-domain access. Services in one domain are provided for users in other domains.

In order to facilitate inter-domain access it is necessary that credentials in one domain are recognised in another domain. Role-Based Trust Management [28] [29] is an example where this occurs. Essentially roles in one domain can facilitate access in another domain where those roles are recognised. Other methods are described in Attribute Based Access Control (ABAC) [42] and Coalition Based Access Control (CBAC) [12].

3.4. Authorization Ordering

A task may be authorized by a number of different people. The concept of "authorization order" recognizes that the authorizations given by one authorizer can override the authorizations of another authorizer. It also recognizes that there is logical order of who to seek an authorization from. Rules can dictate the priority for choosing between multiple authorizers. Various methods of seeking an authorization can also exist. "Discretionary Overriding of Access Control" [36] [35] is an example of work in this area.

3.5. Variable Authorization Timing

Authorization timing has to do with when an authorization is given. While it is normal to give authorizations before access is required, authorizations can also be given at the time access is required or even after access is granted.

Optimistic Security [33] introduced the idea that all accesses can initially be allowed. It allows for integrity to be maintained by providing mechanisms for rolling back data to previous states. Actions can be taken against users who abuse their access rights.

Stevens and Wulf [39] categorised authorizations as *ex ante* when given prior to access, *uno tempore* when given at the time access is required, and *ex post* when given retrospectively.

3.6. Cooperative Workplace Practices

Many access control techniques work well in theory but because of their rigidity they prove difficult to administer in practice. This can be partly due to the fact that they do not take cooperative workplace practices into account. A simple example would be a nurse taking over another nurse's work while he/she is on a break.

To be effective a system must effectively model real world practices. For example, it must be able to:

- Deal with emergencies;
- Deal with unpredictable scenarios;
- Facilitate colleagues helping one another with tasks; and
- Facilitate the delegation of duties.

Many of these problems were dealt with in our previous work on Professional Access Control [15].

3.7. A General Solution

To summarize, the six functions/ criteria that a general solution must handle are:

1) Multiple Access Control Contexts;
2) Interdependence of Tasks;
3) Inter-domain Access;
4) Authorization Ordering;
5) Variable Authorization Timing; and
6) Cooperative Workplace Practices.

Current solutions possess some of these criteria but not all. A cost-effective solution should be general in nature and handle all six criteria.

4. Inbuilt Consent

Privacy compliance requires access control solutions which provide appropriate consent mechanisms. These consent mechanisms must provide functionality that deals with all types of consent.

There are two basic types of consent [11, 13, 24]. They are *express* consent and *implied* consent. Express consent requires that an explicit indication of consent (eg. in writing, electronically or verbally) be given. Implied consent is consent that can be assumed by the circumstances or a person's actions.

There is more to consent than just the acquisition of the consent. For consent to be legitimate it must be given voluntarily and not under any coercion. The client must be properly informed of the implications of their consent being given. The client must also have the capacity to give consent. That is, their mental and physical state must be sufficient for them to be able to give legitimate consent.

4.1. Adding a Consent Mechanism

Role Based Access Control (RBAC) [20] [34] [38] extends the basic Subject-Object model by allowing subjects to be given "roles" which are collections of rights to objects. RBAC itself has been extended to provide additional functionality. TeaM based Access Control (TMAC) [40] [23] [2], as the name suggests, allows users to be grouped into teams and privileges to be given according to team membership. Enterprise RBAC (ERBAC) [21] [19] and other models [8] [14] [18] seek to make role management in large enterprises easier. Rule based RBAC extensions [26] [1] also seek to address complexity problems.

Figure 8 shows this bottom-up approach. The model extensions basically add administrative mechanisms. Consent functionality could be added on top of this by utilizing another layer of administration.

The top-down approach seeks to reduce complexity rather than just adding an extra level of new complexity. To achieve this there is a need to incorporate consent as an integral part of service-related operations. To be acceptable, this must be done in a way that gives productivity and cost benefits.Figure 8 also shows the top-down approach. It uses redefined abstractions based on the common vocabulary to enable consent to be built into the system. The consent mechanism is located in the Consent Management System component of the Integrated System.

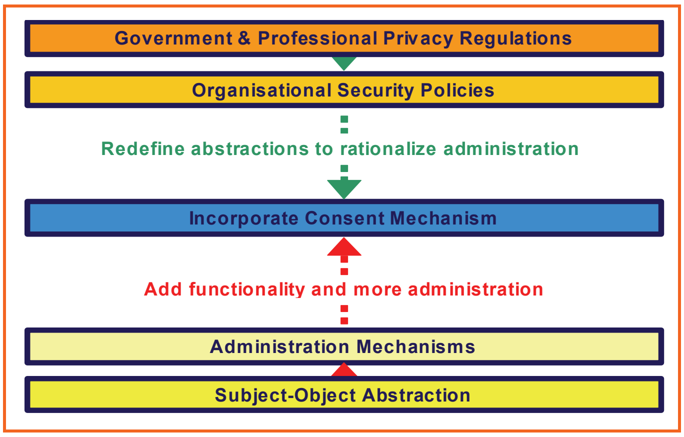
Government & Professional Privacy Regulations
Organisational Security Policies
Redefine abstractions to rationalize administration
Incorporate Consent Mechanism
Add functionality and more administration
Administration Mechanisms
Subject-Object Abstraction

Figure 8: Top-Down and Bottom-Up Views

4.2. The Client-Task Consent Mechanism

The consent concept requires that all tasks that are performed for the *Client* are authorized by the *Client.* In this sense consent is simply an authorization given by the client to the system.

Figure 9 shows the Client-Task consent mechanism.

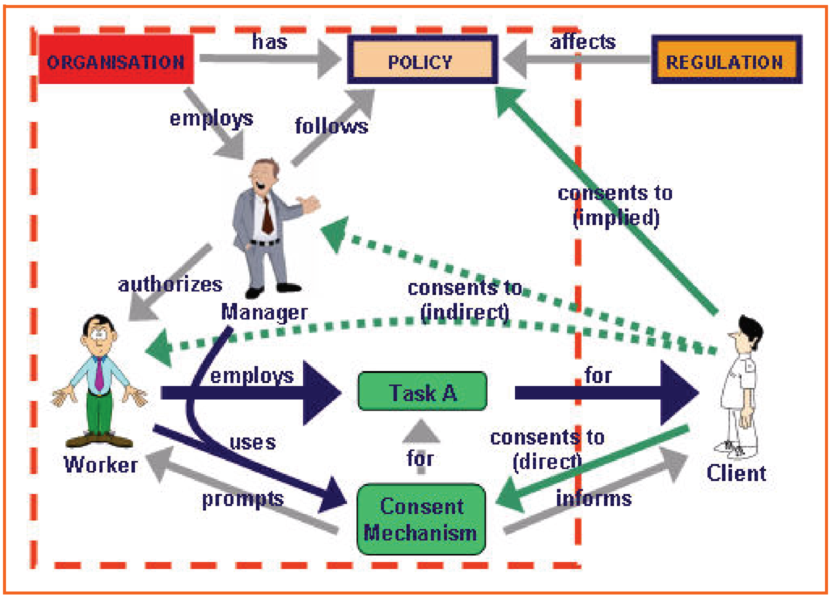

Figure 9: Client-Task Consent Mechanism

The Client-Task system implements *implied* consent by assuming that the *Client,* in agreeing to be serviced by the *Organisation,* accepts the *Policy.* Implied consent is then carried out through the actions of the *Manager.* This can be viewed as an "opt-out" approach, but the existing consent mechanisms can easily be utilized to facilitate an "opt-in" approach.

There are two ways that *express* consent can be given to the system. Firstly, the client can interact with the system to put their authorization directly into the system. This is termed *direct* consent. Secondly, the client can give written, verbal or electronic consent to either the *Manager* or the *Worker* who then must enter the authorization into the system. This is termed *indirect* consent.

The *Consent Mechanism* is the part of the system that requires and collects the consent authorizations. Consent authorizations can each be implemented through a task which is associated with the primary task. The consent task can incorporate functionality to prompt the worker about what he must do, as well as to inform the client about the implications of their consent.

5. Key Concepts

This section discusses the key concepts raised in this paper. These concepts can be attributed to taking a top-down approach to the problem. The top-down approach allows us to consider the access control abstractions in terms of compliancy and workplace requirements rather than the traditional way of treating the abstractions as technical "building blocks".

5.1. System Compliance

By building a consent mechanism into the system, proof of regulatory compliance may be provided. Proof lies in the fact that when the consent mechanism is built into the system so that it is automatically triggered, consent based tasks can be tracked and audited. The system can prompt workers and require them to follow compliant procedures. This can limit the number of mistakes and oversights that occur. The ultimate result is that an ability to demonstrate compliance to privacy legislation may ensure that litigation is avoided. Compliant systems may thus lead to substantial reductions in insurance costs, which is a major issue in the health sector.

5.2. The Concept of Clients

Discretionary Access Control (DAC) models [30] allow users to specify how other users may access the objects they own. In organisations where the clients are not system users, yet have a level of ownership over their records, there is no direct mechanism for clients to specify the access requirements to the objects they own. Rather, clients are at the mercy of the organisation to correctly control access to their records.

The fundamental reason why our approach differs from previous approaches is that it inherently recognises that the *Owner* of the data is not usually a user of the system. The dashed red square in Figure 5 represents the boundary of the *Organisa-*

tion. The *Client/Owner* is outside the *Organisation* while the *Worker* and the data are inside. This models a standard relationship between organisations, workers and clients. The top-down approach led to this differentiation between workers and clients.

5.3. Dependency Between Rights

There is perhaps some confusion in the security community about what constitute static access control and dynamic access control. It is sometimes stated that static access control refers to rights that are specified in advance and tend not to change and dynamic access control is applicable when the context of an access is checked at the time of the access.

We believe that it is more useful to think in terms or whether an access right or a task is "dependent" or "independent". The real reason why Subject-Object based systems are seen as static is that they do not express any dependency between access rights. While the use of sessions and mutually exclusive roles [38] does build a level of dependency into systems, it is questionable whether these dependencies should be at the level of roles. Rather, we believe that roles must be independent and that it is tasks/duties that should be dependent. For example, when a restricted medication is prescribed in a hospital which requires the signatures of two nurses before being administered, the two nurses do not change their roles. Rather there are two tasks, one of signing and the other of countersigning, either of which can be performed by a nurse. It is vital not to confuse tasks and roles or to model tasks as roles.

5.4. The Use of Roles

It is our view that roles should be viewed as just one type of context grouping. Obviously they are the most important and useful type of worker grouping, but they are not different in nature from other worker groupings. While most systems need to employ access controls which take roles into account, they do not have to. For example, access could be controlled according to the worker's network address or location.

In the Client-Task approach the term role refers to an organisational position. A role is what a person *is*. A duty or task is what they *do* (in the given role). The Client-Task approach also sees the question of who is given a role (or put into any group) as a management decision. The questions of what rights are given to a task and what tasks are assigned to a role (or group) are seen as decisions for system administrators.

6. Conclusions and Further Work

The protection of low-value client information has proven difficult in the health industry. This is because solutions must be low-cost while at the same time being compliant with complex regulations. It was argued that for low-cost compliance to be achieved an *integrated general system with inbuilt consent* is the principal requirement. The Integrated System based on the Client-Task approach that we propose in this paper meets this requirement.

The inbuilt consent mechanism used in the Integrated System utilizes consent tasks which are triggered by service-related tasks. This enables security to be incorporated with positive benefits rather than just the imposition of extra work. Integration and generalisation thus provide productivity and usability benefits which allow increased automation and feedback to workers. In addition, compliance may lead to substantially reduced insurance costs.

While it is technically possible to add consent mechanisms to existing Subject-Object based system such as RBAC, it is our contention that such bottom-up approaches can never be practically feasible. This is because such added mechanisms generate significant and costly overheads and can never guarantee privacy compliance because privacy regulations and policies are not expressed in terms of subjects, rights and objects.

The Integrated System we propose integrates the technical, management and regulatory components of the system by using the Client-Task approach. This approach introduces the concept of the *Client* to access control. The concept assumes the Client has some form of ownership over their personally identified data and gives them the required control over it even though they are not users of the system. The Client-Task approach also sees the question of who is put into a group as a personnel management decision, while the question of which rights are given to a group as a decision for system administrators.

It was found that for the system to be general in nature it must:

- Handle rules with multiple contexts;
- Represent constraints through task dependencies;
- Enable inter-domain access;
- Handle authorizations in an ordered fashion;
- Allow authorizations before, at the time of and after access is made; and
- Facilitate cooperative work practices.

Further work involves the development and testing of a comprehensive Integrated System in various organizational settings. The primary scenario will be a hospital as this has a high level of complexity. Different ownership and joint ownership variations will also be considered. Data storage options are another related area which may be investigated.

Regulations are developed outside an organisation. In current systems policy must be manually changed to reflect regulation. It may be feasible however, to have a system where any changes to regulations can dynamically change policy. This would be similar to current techniques which are used to automatically update software and virus definitions.

References

1. Al-Kahtani, M.A. and Sandhu, R., A Model for Attribute-Based User-Role Assignment. in 18th Annual Computer Security Applications Conference, (Las Vegas, Nevada, USA, 2002), IEEE, 353.

2. Alotaiby, F.T. and Chen, J.X., A Model for Team-based Access Control (TMAC 2004). in International Conference on Information Technology: Coding and Computing (ITCC'04), (Las Vegas, Nevada, USA, 2004), IEEE.

3. Atluri, V. and Warner, J., Supporting Conditional Delegation in Secure Workflow Management Systems. in Symposium on Access Control Models and Technologies 2005, (Stockholm, Sweden, 2005), ACM Press, New York, NY, USA, 59-66.

4. Bacon, J., Moody, K. and Yao, W. A Model of OASIS Role-Based Access Control and Its Support for Active Security. ACM Transactions on Information and System Security, Vol. 5 (No. 4). 492–540.

5. Beresnevichiene, Y. A role and context based security model, University of Cambridge Computer Laboratory, Cambridge, 2003.

6. Bertino, E., Ferrari, E. and Atluri, V. The Specification and Enforcement of Authorization Constraints in Workflow Management Systems. ACM Transactions on Information and System Security, Vol. 2 (No. 1). 65–104.

7. Botha, R.A. and Eloff, J.H.P. Separation of duties for access control enforcement in workflow environments. IBM Systems Journal, 40 (3). 666-682.

8. Caelli, W. and Rhodes, A. RBACManager: Implementing a Minimal Role Based Access Control Scheme (RBACM) Under the Windows NT 4.0 Workstation® Operating System, 1999.

9. Chen, F. and Sandhu, R.S., Constraints for role-based access control. in Symposium on Access Control Models and Technologies, (Gaithersburg, Maryland, US, 1996), ACM Press, New York, NY, USA.

10. Clark, D.D. and Wilson, D.H., A Comparison of Commercial and Military Computer Security Policies. in IEEE Computer Society Symposium on Security and Privacy, (Oakland, USA, 1987).

11. Clarke, R., e-Consent: A Critical Element of Trust in e-Business. in 15th Bled Electronic Commerce Conference, (Bled, Slovenia, 2002).

12. Cohen, E., Thomas, R.K., Winsborough, W. and Shands, D., Models for Coalitionbased Access Control (CBAC). in Seventh ACM symposium on Access control models and technologies, (Monterey, California, USA, 2002), ACM Press, 97-106.

13. Coiera, E. and Clarke, R. e-Consent: The Design and Implementation of Consumer Consent Mechanisms in an Electronic Environment. Journal of the American Medical Informatics Association, 11 (2). 129–140.

14. Crook, R., Ince, D. and Nuseibeh, B., Towards an Analytical Role Modelling Framework for Security Requirements. in 8th International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ-02), (Essen, Germany, 2002).

15. de la Motte, L. Professional Access Control School of Computing, University of Tasmania, Launceston, 2004.

16. Desmond, J. Roles or Rules: The Access Control Debate, esecurityplanet, 2003.

17. El Kalam, A.A., Baida, R.E., Balbiani, P., Benferhat, S., Cuppens, F., Deswarte, Y., Miege, A., Saurel, C. and Trouessin, G., Organisation based access control. in 4th International IEEE Workshop on Policies for Distributed Systems and Networks, (Lake Como, Italy, 2003), IEEE, 120-131.

18. Fernandez, R. Enterprise Dynamic Access Control (EDAC) Overview, SSC San Diego, 2005.

19. Ferraiolo, D. Evolution of Access Control in Commercial Products, 2003.

20. Ferraiolo, D. and Kuhn, R., Role-Based Access Control. in 15th National Computer Security Conference, (Baltimore, MD, 1992).

21. Ferraiolo, D.F., Ahn, G.-J., R.Chandramouli and Gavrila, S.I., The Role Control Center: Features and Case Studies. in 8th ACM Symposium on Access Control Models And Technologies, (Como, Italy, 2003), ACM Press New York, NY, USA, 12 - 20.

22. Fischer-Hubner, S. and Ott, A., From a Formal Privacy Model to its Implementation. in 21st National Information Systems Security Conference, (Arlington, VA, 1998).

23. Georgiadis, C.K., Mavridis, I., Pangalos, G. and Thomas, R.K., Flexible Team-Based Access Control Using Contexts. in SACMAT '01, (Chantilly, Virginia, USA, 2001), ACM, 21-27.

24. HealthConnect. Consent and Electronic Health Records - A Discussion Paper, 2002.

25. Hung, P.C.K. and Karlapalem, K., A Secure Workflow Model. in Australasian Information Security Workshop (AISW2003), (Adelaide, Australia, 2003), Australian Computer Society, Inc. - Conferences in Research and Practice in Information Technology.

26. Kern, A. and Walhorn, C., Rule Support for RoleBased Access Control. in Symposium on Access Control Models and Technologies 2005, (Stockholm, Sweden, 2005), ACM Press, New York, NY, USA, 130-138.

27. Lampson, B.W. Computer Security in the Real World, 2002.

28. Li, N. and Mitchell, J.C., Design of a Role-based Trust-management Framework. in IEEE Symposium on Security and Privacy, 2002, (2002), IEEE.

29. Li, N. and Mitchell, J.C., RT: A Role-based Trust-management Framework. in Third DARPA Information Survivability Conference, (2003).

30. NCSC. A Guide to Understanding Discretionary Access Control in Trusted Systems (Neon Orange Book), 1987.

31. Neumann, G. and Strembeck, M., An Approach to Engineer and Enforce Context Constraints in an RBAC Environment. in SACMAT '03, (Como, Italy, 2003), ACM, 65-79.

32. OECD. OECD Guidelines on the Protection of Privacy and Transborder Flows of Personal Data, 2006.

33. Povey, D. Optimistic Security: A New Access Control Paradigm, 1999.

34. Rhodes, A. and Caelli, W. A Review Paper Role Based Access Control, University of Queensland, Brisbane Australia, 1999.

35. Rissanen, E., Firozabadi, B.S. and Sergot, M. Discretionary Overriding of Access Control in the Privilege Calculus, 2005.

36. Rissanen, E., Firozabadi, B.S. and Sergot, M. Towards A Mechanism for Discretionary Overriding of Access Control, 2004.

37. Russell, N., Hofstede, A.H.M.t., Edmond, D. and Aalst, W.M.P.v.d. Workflow Resource Patterns, 2005.

38. Sandhu, R.S., Coynek, E.J., Feinsteink, H.L. and Youmank, C.E. Role-Based Access Control Models. IEEE Computer, 29 (2). 38-47.

39. Stevens, G. and Wulf, V. A New Dimension in Access Control: Studying Maintenance Engineering across Organizational Boundaries, 2002.

40. Thomas, R.K., Team-based Access Control (TMAC): A Primitive for Applying Role-based Access Controls in Collaborative Environments. in RBAC '97, (Fairfax Va USA, 1997), ACM, 13-19.

41. Thomas, R.K. and Sandhu, R.S., Task-based Authorisation Controls (TBAC): A Family of Models for Active and Enterprise-oriented Authorisation Management. in IFIP WG11.3 Workshop on Database Security, (Lake Tahoe, California, USA, 1997), Chapman & Hall.

42. Wang, L., Wijesekera, D. and Jajodia, S., A Logic-based Framework for Attribute based Access Control. in 2004 ACM workshop on Formal methods in security engineering, (2004).

Correspondence

Leigh de la Motte
School of Computing
University of Tasmania
Locked Bag 1359
Launceston TAS 7250
Email: lhdela@utas.edu.au