OASIS role-based access control for electronic health records

D.M. Eyers, J. Bacon and K. Moody

Abstract: The United Kingdom National Health Service strategy is to have a full electronic health records service available nationally by 2008. Secure, scalable, access control infrastructure will be critical to the success of such an initiative. To this end, a prototype for such a service, built over the OASIS role-based access control architecture is presented. How the implementation achieves scalability is explained and the complex policy requirements are discussed.

1 Introduction

Providing electronic health records (EHRs) with high availability, yet maintaining their protection from unauthorised access, is a complex yet crucial task. One goal published within the 1998 United Kingdom National Health Service (NHS) information technology strategy documents [1] was to achieve EHR support nationally by 2005 (or earlier). This highly ambitious target was revised in 2002 to suggest national EHR support by 2008. Even so, we feel it is unlikely that the full scope intended of NHS EHRs will be achieved by then.

Access control in EHR systems poses a number of challenges not faced in other security environments. For a start, the information being protected is highly personal–security breeches may lead to irrevocable consequences for the individuals involved. Yet, at the same time, there is a need in emergencies to access all the information relevant to the conditions of a patient (e.g. their allergies to certain medication, their HIV status, and so on). In addition, aggregated statistics are necessary to inform strategy within the NHS–for example, the cancer registry monitors the frequency and distribution incidence across the country.

Another difficulty is heterogeneity, both within management and at the system level. By the former, we mean that different health care organisations (HCOs) will all have slightly varying procedures, and thus policies dictating access control. Of course, these local differences will need to be balanced with the overall directives of the NHS.

Perhaps the most challenging aspect of the NHS EHR strategy, however, is the amount of power given to the patient in terms of access control. 'Patient consent and confidentiality' are both considered key issues relating to the NHS EHRs. This amounts to patients being able to specify fine-grained access control restrictions over their individual EHRs if they so choose, potentially leading to a large, distributed, complex policy.

To further complicate matters, and increase the need to manage policy conflicts, there are various situations in which health professionals will need to apply emergency overrides. In simple terms, patients may unknowingly specify access control rules that are not in their own best interest, and may indeed turn out to be life-threatening. Clearly it is preferable that the access control policy should incorporate emergency information access needs directly, although it is unlikely to be easy to identify all the data that might be relevant to any potential emergency situation.

This paper describes our research into security for EHRs and the development of an NHS EHR prototype. We cannot claim to have solved all of the above issues, but we believe our software is sufficiently flexible and expressive to do so, and the experience of developing a prototype has been valuable.

Our architecture was designed in collaboration with the Cambridge-based company Clinical and Biomedical Computing Limited (CBCL), and takes a role-based approach to security management. It is built over the Open Architecture for Secure Interworking Services (OASIS, see [2–4]), an established role-based access control (RBAC) model developed at the Cambridge Computer Laboratory. The prototype software was developed by CBCL, and hence is referred to as CBCL OASIS for the rest of this document.

2 Background and related work

This Section discusses research related to the provision of access control for EHR infrastructures. We begin by providing an overview of RBAC, then examine some of the research projects that directly investigate the problem of EHR access control.

2.1 Role-based access control

Mandatory access control (MAC) is well suited to military-style applications where there is a strict ordering to both principals (e.g. users) and privileges. Strong guarantees hold over any given security state. The Bell LaPadula model [5] is one of the fundamental MAC proposals (see [6] and [7] for extensions to it).

For many applications, including EHR management, MAC is simply too restrictive [8]. Discretionary access control (DAC) schemes offer much more flexibility. They generally represent an access control matrix indicating

IEE Proceedings online no. 20045038

doi:10.1049/ip-sen:20045038

Paper first received 7th October 2004 and in revised form 17th August 2005

The author is with the University of Cambridge Computer Laboratory, JJ Thomson Avenue, Cambridge, UK

E-mail: david.eyers@cl.cam.ac.uk

which subjects (one row for each) can access which objects (each column) via which modes (the cell contents). It is usually most convenient to separate the storage of rows or columns. A capability-based system issues the set of objects a subject may access to that subject. Conversely, an access control list stores the subjects that can access an object with that object (for an overview see [9]).

The drawback with DAC schemes is their lack of manageability based on this splitting up of the access control matrix–for example, addition and deletion of users or protected resources requires discovery and treatment of all dependent entries.

Role-based access control (RBAC) [10, 11] is designed to simplify security administration by introducing the 'role' abstraction (which was mentioned as early as in [12]) between principals (subjects) and privileges (objects). This splits management of the principal to privilege mapping by splitting it into two parts: a mapping from users to roles, and a mapping from roles to privileges. In [13] (and soon to be an ANSI standard), four RBAC schemes are described. $RBAC_0$ simply dictates that there are user–role and role–privilege relationships. $RBAC_2$ is the most closely related to the OASIS system. $RBAC_2$ extends the basic $RBAC_0$ model by adding role–role relationships, as shown in Fig. 1. Each role–role relationship (labelled as 'Rules' in Fig. 1) can be thought of as a directed edge between roles, and has an associated constraint that must be satisfied if a user is to activate the target role based on their already being active in the source role. $RBAC_2$ facilitates the deployment of powerful policy schemas, two such examples being cardinality constraints, and separation of duties constraints [14]. In the former, we restrict the number of users who can be active in a certain role. In the latter we divide roles (and thus privileges) into mutually exclusive sets.

Beyond its administrative benefits, roles also often intuitively model human functions within an organisation. For example, roles may be defined that well match the positions in an organisational hierarchy. Alternatively, 'functional roles' aim to model tasks performed by subjects in an organisation. The process of determining the roles required to model a security application is often referred to as 'role engineering' and is well explored in the literature [15–20].

2.2 Expressive policy languages: Cassandra

A recent development in policy languages relating to EHR management is Cassandra [21]. It expresses policy using datalog clauses augmented with constraints from a given constraint domain. Becker has defined a number of constraint domains, each having increasing expressiveness, but predictably coupled with increasing complexity for evaluation. In all cases they provide proven bounds on the time until inference process termination, which is an essential safety property when computing security policy predicates. Cassandra was evaluated by showing it could encode the NHS Patients' Charter [22].

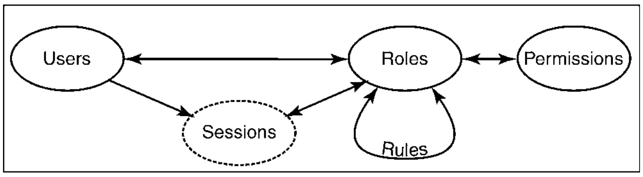


Fig. 1 *Users, roles, sessions, privileges and constraints in* $RBAC_2$
Note that only one user can hold any given session. All other relationships are many to many

The main security-specific aspect of Cassandra involves its role awareness; a number of specific predicates marshal role activation and deactivation. `canActivate` and `canDeactivate` are guards for the assertion and removal of `hasActivated` and `isDeactivated` clauses, each of which specify a principal and a role. Further, Cassandra includes the `canReqCred` predicate–if true a given principal can request a credential from a remote service.

Owing to its very general policy language, it supports the specification of complex policy including role hierarchies, separation of duties, delegation, and various types of revocation, although Becker has pointed out that most of these complex policy features were required infrequently in his evaluation.

There are numerous other established policy languages with high expressiveness, for example, Ponder [23] and Rei [24]. These languages provide support for additional modality including obligations and prohibitions. However, neither has been used directly in EHR applications.

2.3 Managing conflicts: Tees Confidentiality Model

As mentioned in Section 1, one of the complications facing access control for EHRs is the need to support emergency overrides under certain circumstances. The Tees Confidentiality Model (TCM) [25–27] provides a specific ordering designed to manage conflicts consistent with the proposed UK Patient Confidentiality Requirements.

Confidentiality permissions are processed in a defined order connected with specific types of policy from least to most powerful (in terms of override) labelled: IRC, ISC, IGC, RSC, and RGC (the letters I, R, S, G and C representing Identity, Role, Specific, General and Confidentiality, respectively). The idea is that these stages cover all the useful configurations for positive and negative permissions levied over individuals, groups of individuals (e.g. teams), roles, groups of roles, particular data records, and groups of data records.

Because the TCM is very specifically engineered to the medical context, it is likely that its features could easily be included in other access control technologies. Although it includes negative permissions, which complicate authorisation inference, it does so in a strictly limited manner within the above levels.

2.4 Distributed policy: PERMIS

The PERMIS (PrivilEge and Role Management Infrastructure Standards validation) distributed access control architecture [28, 29] is another system that has been applied to problems in the health care domain [30]. It is tightly coupled with the ISO/IEC 10181-3 standard (the Access control framework section of the OSI security framework).

PERMIS identifies principals through the use of X.500-style distinguished names. It is agnostic to the particular mechanism used to connect a principal with a distinguished name. A common approach is to use PKI (public key infrastructure) to form this linkage via X.509 identity certificates.

The actual attributes on which access control decisions are based, however, are recorded in a different type of X.509 certificate: Privilege Management Infrastructure (PMI) certificates. These credentials are linked to the identity of a principal through the public keys of the X.509 identity certificates, separating the 'permanent feature' of identity from 'changeable attributes' relating to authorisations. To manage these collections of X.509 certificates, PERMIS uses the Lightweight Directory Access Protocol (LDAP).

PERMIS uses PMI certificates for its policy representation. These policy certificates indicate the prerequisites needed to acquire some given privilege. The advantage of again using certificates for policy is that they too can be stored in the existing LDAP directories, and it allows different organisations to define their own policy fragments, avoiding the need for centralised policy maintenance. A comparison between PERMIS and OASIS is provided in [31].

2.5 EHR management

There have been many architectures proposed to handle EHR systems. One which does not focus on security, but has a similar network architecture to our EHR prototype, is Synapses [32].

The Patient Centered Access to Secure Systems Online (PCASSO), [33] project is a trial EHR system run in the United States and including Sun Microsystems in its later stages of trial. Although it claims to be role-based, its role classifications are very basic. Overall, the project focused on security risks in the computing infrastructure (e.g. web-browsers logging keystrokes and data) rather than the complex policy issues required to support patient control over patient records.

3 OASIS

The Opera Research group at the University of Cambridge Computer Laboratory developed the Open Architecture for Secure Interworking Services (OASIS) [34, 35] as an extended RBAC model. It has parameterised policy rules based on a Horn-clause logic, and provides a flexible environmental interaction mechanism.

OASIS roles are activated in the context of sessions. Role activation rules check all conditions are satisfied before roles are activated within a session. Authorisation rules then perform further checks before privileges are exercised on a protected system. OASIS roles may carry parameters to allow fine-grained decisions to be made on a subgroup of a role if desired. OASIS rules are parameterised also, and may include environmental predicates to check conditions external to the OASIS access control framework.

Each of the distributed entities for which OASIS provides access control is wrapped by an OASIS service, which itself may be distributed over OASIS servers. These services all operate in an asynchronous manner, and send messages to each other to maintain a consistent distributed state.

The concept of 'appointment' [34] in OASIS generalises delegation in other RBAC systems – certain privileges are designed to create and revoke credentials that are used as role activation prerequisites by other users. Credentials that need to persist beyond the duration of an OASIS session are called 'appointment certificates', and can be used as prerequisites in role activation rules.

The basic structure of a role activation rule is as follows:

$$r_1, r_2, \ldots, r_{n_r}, ac_1, \ldots, ac_{n_{ac}}, e_1, \ldots, e_{n_e} \vdash r$$

The r_i, ac_j and e_k terms represent the n_r prerequisite roles, n_{ac} appointment certificates and n_e environmental constraint predicates in this rule, respectively–note that it is acceptable for any of n_r, n_{ac} or n_e to be zero, provided at least one is non-zero. Predicate expressions on the left-hand side of the rule are called preconditions, and must be valid for a given user to activate r, the target role. Roles and appointment certificates are valid if they have not been revoked. Environmental predicates are valid if they evaluate to be true.

Authorisation rules are of the following form:

$$r, e_1, \ldots, e_{n_e} \vdash p$$

There is one and only one prerequisite role r. The environmental constraints e_k behave as for role activation rules, and finally p is the target privilege of this rule. A set of the above role activation and authorisation rules defines the policy for a given OASIS service. Note that in the CBCL OASIS implementation we use an XML representation of the above policy language.

4 An electronic health record infrastructure

In this Section we describe the design of our NHS EHR infrastructure, and discuss how the prototype implementation demonstrates its applicability. The subsections following describe each of the main design considerations.

4.1 Heterogeneous services

In an ideal world, requirements analysis could be performed over the entire EHR network, an implementation written, and health care organisations (HCOs) be required to change over to it. Of course this is completely unrealistic, if only because the requirements of Health Care data protection will always continue to evolve.

Given that requirements are constantly changing, it is then problematic that financial and time constraints mean that different parts of the larger HCO network update their software without any global synchronisation. Thus, in designing our EHR infrastructure, we assume that we will have to integrate heterogeneous data sources.

One particular advantage of the NHS context, however, is that certain fields, such as NHS identifiers, are almost invariably present in a semantically consistent manner in EHRs, even if the semantics of these records and the correlations between them require high-level medical training. Indeed this is the basis of our proposed 'multi-window approach', a term introduced by CBCL director Jem Rashbass during the design process. In essence our approach involves EHR fragments being retrieved from different sites, and the joining of important information being done by qualified practitioners seeing these fragments, rather than by an automated process. So we support the management of heterogeneous sources by retaining an active human role in the merging of the data they may individually present.

Our intuition is that by using simple attributes such as the NHS ID of the patients and those of the practitioners treating them, combined with the time-stamp of the event in question, an index can be formed over EHRs without the need to become involved in the semantics of the event in question. This then avoids the need to define a centralised data dictionary of terms and messages.

Another advantage of this thin indexing approach is that it will allow a global index to be formed that needs to maintain very little data per record. A URI references the bulk of the medical data at the hosting HCO sites. Given that the index should span all active records, it is desirable that the index have minimised resource requirements.

One disadvantage of our approach is that, because there is a two-step process in retrieving record fragments, the access control policy cannot be specified at the level of data used in the multi-window retrieval, and instead needs to be levied at the HCO EHR fragment level. Although administration of the policy becomes more complex, it is easier to guarantee that access to particular EHR fragments will be denied. More difficult is ensuring the availability of aggregated results. Note that it is also possible to use the index service as a point of anonymity support, because it can (and does) broker a transaction between a client and an HCO using pseudonyms.

The hierarchical nature of our design also resonates well with the way policy is likely to be specified within the NHS. For example, there will be high-level policy directives connected with the Integrated Care Records Service (ICRS) that operates over the entire NHS. Yet each individual HCO will need to tune policy to suit their specific organisations. Thus, in many cases, higher-level policies can be specified closer to the global EHR index.

To avoid having to micro-manage the different aspects of a heterogeneous HCO network, we impose an overall network infrastructure, but require each service to only provide interoperation on a limited set of terms. The idea is that these terms can facilitate opening channels to the HCOs, which show web-based EHR fragments to a medically qualified user, and thus avoid the need for complete standardisation of the data dictionaries in use at each HCO. We are hopeful this approach will best cope with the heterogeneous data that electronic health record management will continue to present for the foreseeable future.

4.2 An NHS EHR infrastructure

The main components of the CBCL OASIS prototype are presented in this Section, and are illustrated in Fig. 2. It is important to note that only the Index Service is a single logical object, although we expect it would be replicated for the sake of availability and reliability. Each of the types of node in the figure are described in the following.

HCO server. It is assumed that each health care organisation needs to have an OASIS-aware service that allows it to interact with the rest of the EHR network. If a legacy application is being used at a given site, the OASIS-aware service must define sufficient local policy to allow this application to participate in the wider EHR network. Given HCO servers contain (possibly through a wrapper) the detailed EHR information, the most semantically rich access control policies will be performed by the HCO servers.

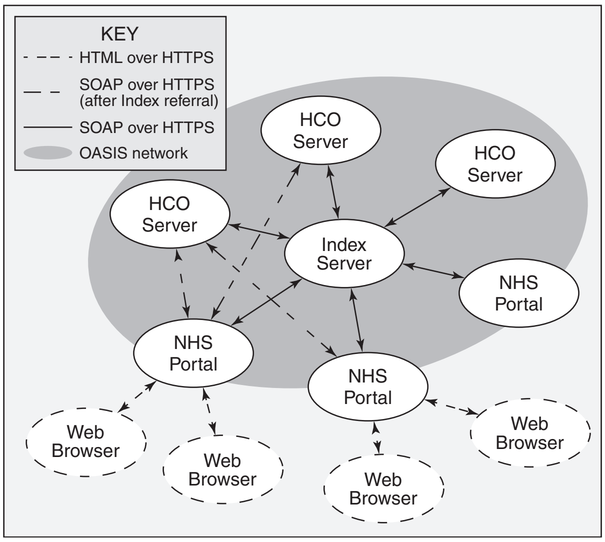


Fig. 2 *EHR network components*

Index service. The function of the index service is implied by its name. It stores a rudimentary header for each EHR fragment, and which HCO to contact for the complete record. It also translates the source of requests for EHR fragments into pseudonyms before they reach the HCO sites in question. Note that it is necessary for the index service to contain access control policy. For example, certain situations may require that even the EHR headers are protected from certain requesters, rather than access control being performed at the HCO sites after it is known where the EHR fragments reside. For example, some HCOs may, through their name alone, imply sensitive information, such as psychiatric institutions, or HCOs related to blood testing or HIV/AIDS. In such cases index server policy might filter responses based on the NHS ID of the medical practitioner.

NHS Portal. The NHS portal provides the connection to the user side of the EHR network, where users can be either patients or medical staff. The current prototype provides an HTTPS (i.e. secure) web-serving ability so that users' terminals need only run a web browser to allow them to use the EHR network (which is an OASIS service network). Numerous users can be managed by any given NHS portal, and NHS portals themselves can be replicated to provide increased efficiency (e.g. geographical localisation), and increased reliability. Again, the NHS portal may also contain OASIS policy. This avoids expensive rejection by the HCOs in cases where a fairly obvious policy breach is being submitted. Doctors will have two main roles within the EHR network – as health professionals, and as patients themselves. The requests they can legitimately make of their own patients will usually not be valid from the perspective of other patients. NHS portal policy can enforce simple access control roles to catch a situation in which a doctor has accidentally made an EHR request from the role of 'patient' rather than 'doctor'.

When analysing communication between the above components, it is important to realise that there are both internal and external interactions from the perspective of CBCL OASIS. The internal communications occur between the software at HCOs (which might be wrappers over legacy applications), the index server, and the NHS portal. Internal communication transmits SOAP messages [36] over HTTPS connections. The external communication between users' web browsers and the NHS portal currently uses conventional HTML over HTTPS.

5 CBCL OASIS

To put the interactions between the various CBCL OASIS components (all of which are OASIS-aware services) into context, we provide a trace of the overall steps taken between a user making a request for EHR data, and a formatted response arriving back at that user. The steps in this communication are shown in Fig. 3, and are described in the following. A number of extensions to OASIS were implemented during our EHR trial to support anonymity and informed decisions regarding caching of credentials, but due to a lack of space we do not discuss these extensions in this paper.

Note that before users can communicate with the NHS portal, they must have been issued with an X.509 authentication certificate. Because this step is likely to take place once, long before users make EHR requests, and because

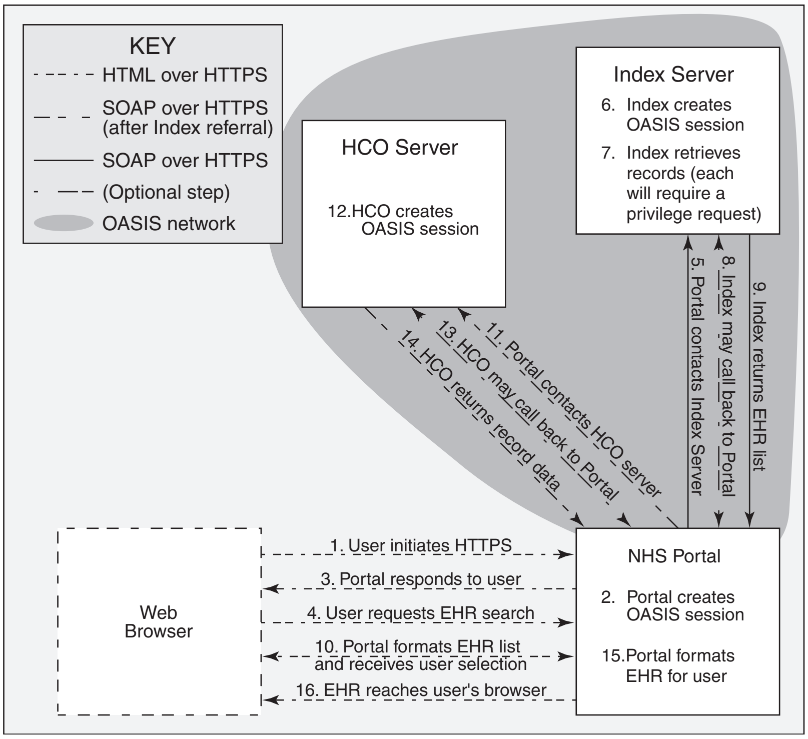

KEY
HTML over HTTPS
SOAP over HTTPS (after Index referral)
SOAP over HTTPS
(Optional step)
OASIS network
HCO Server
12.HCO creates OASIS session
Index Server
6. Index creates OASIS session
7. Index retrieves records (each will require a privilege request)
5. Portal contacts Index Server
8. Index may call back to Portal
9. Index returns EHR list
11. Portal contacts HCO server
13. HCO may call back to Portal
14. HCO returns record data
Web Browser
NHS Portal
2. Portal creates OASIS session
15.Portal formats EHR for user
1. User initiates HTTPS
3. Portal responds to user
4. User requests EHR search
10. Portal formats EHR list and receives user selection
16. EHR reaches user's browser

Fig. 3 *Retrieving an EHR using the CBCL OASIS NHS prototype*

the distribution will be out-of-band with respect to OASIS, we have not included it in the list of steps below.

The user's X.509 certificate contains information used as the initial appointment within the user's OASIS session. Initial role acquisition is thus split into two phases: opening an initial connection, and providing initial credentials. For a communication channel to be able to be opened, there must be a shared trusted third party – namely, a common certificate on the certificate chain used by the client and the NHS portal. Extension fields within this X.509 certificate allow the NHS portal to issue a starting role within this particular user session. The CBCL OASIS prototype includes a web-based system for issuing certificates.

There are two broad categories of access we would expect to the NHS portal: people retrieving their own health records, and medical or administrative staff retrieving EHR fragments to which they have some official connection. In the example below we trace through the interaction we would expect for a patient. We use 'patient' or 'user' to indicate the human user involved in this interaction, and 'client' to indicate their computer. The 'server' being discussed should be clear from context.

1. Our patient decides they wish to examine one of their EHRs. The first step they take is to open their web browser, and direct it to the NHS portal site. This site uses HTTPS security. Note that, unlike most e-commerce sites, the underlying SSL connection requires both client- and server-side authentication. The server-side authentication is standard; the NHS Portal in this case expects to need to present its server certificate. Client-side trust is established automatically by the client-browser if the server's certificate was issued by one of the root or intermediate certification authorities preloaded into the user's browser. Otherwise, the explicit interaction is required by the user – they can choose to import the issuing certificate into their root and intermediate authority list.

Often the client-side authentication will be done automatically if the client only has one suitable certificate in their browser's key-store. Otherwise, the user is usually asked by their browser to select from a list of the certificates they have available. By 'suitable' we mean that X.509 configuration fields in the certificate indicate it is permissible to use it for authentication.

In the cases where the user is not a patient, it is likely that they would see client-certificates for both their role as a doctor or administrator, and also for their role in the NHS as a patient. Note that these multiple certificates could be replaced by an identity certificate, and the policy file use other techniques (such as database lookup) to present them after authentication with a choice of role. It has been observed that the CBCL OASIS implementation is currently blurring X.509 identity and attribute functions in the current OASIS X.509 certificates. There is definitely identity/attribute overloading at the moment, although it could easily be addressed if required.

2. After successful mutual client/server SSL authentication, the NHS portal site creates an OASIS session. In CBCL OASIS this involves the creation of an Enterprise JavaBean (EJB), which is added into the back-end store so that server interruptions (e.g. power loss) will not cause the session state to be lost. In particular, the SQL table used to make the EJB state persistent contains a long integer identifier for this session. We refer to this ID as an OASIS token. The token is essentially random, in that there is no explicit correlation between the user and the token.

In the current CBCL OASIS implementation, because there is no way to know whether the user will make further requests of the NHS portal within this session, a session timer is also initialised. When this timer expires, the session is automatically removed. Further interactions from the user may warrant resetting the timer to extend the maximum session duration.

3. Having initialised the OASIS session state, the Portal sends a response back to the user. Because this response is generated on a per-user basis, it can be personalised

appropriately for them. In the CBCL OASIS EHR prototype, certain links are added to a navigation menu depending on whether certain privileges were able to be granted to the user based on their role. This is shown in Fig. 4 – the left display represents the public view of a page, whereas on the right the privileged ‘HRI Access for NHS patients’ link has been revealed. The decisions leading to granting or denial of privileges occur within the portal, based on its policy file. In the case of our user, they must successfully satisfy the preconditions of the OASIS role activation rule to enter the ‘patient’ role on the portal.

4. It may be the case that the user wishes to browse pages that do not require authorisation based on their active roles. In these cases, the SSL connection provides a secure channel, but the dynamically generated pages from the server will not include any privilege checks.

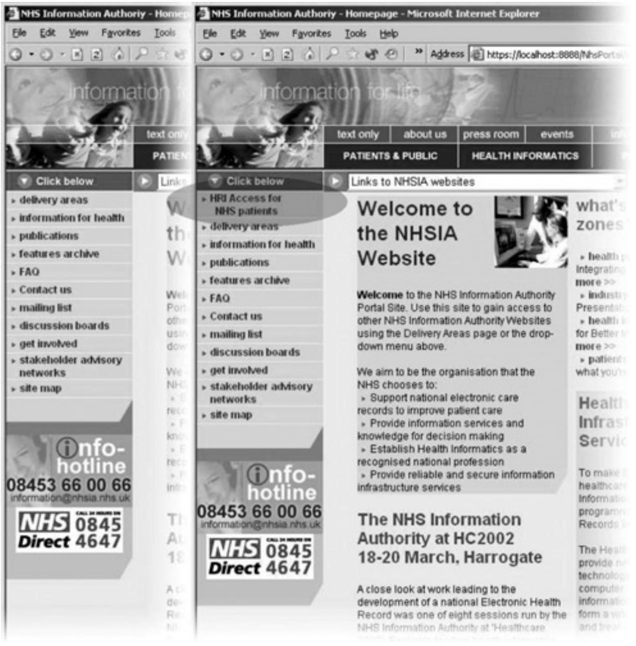
In our example we assume that the user requests to see their patient records, via the customised link provided in the previous step, however. Were the user to be a healthcare professional, this link would take them to a page from which they could select EHRs to examine for which they are directly responsible.

5. On seeing this request for EHRs, the NHS portal contacts the index server. This communication is internal to the OASIS network, and occurs using SOAP over HTTPS. Again the HTTPS connection between these two nodes uses two-sided authentication; this time the server certificates of the portal and the index server.

The portal provides the user token within whose session this request has been generated.

6. After successful connection establishment with the Portal, the Index Server Creates its own OASIS session and associated token. As for the portal, the critical state for this session is written to a persistent back-end database. The structure of the index server sessions are almost identical to those of the portal, because both use the same code base for session management – that is, instances of the `OasisSessionServer`. Unlike the portal sessions, however, the index sessions also record a link field which contains the portal token value.

The portal token value is maintained in order to facilitate the index server asking further questions of the portal regarding the credentials of the user making the current request. This will be necessary when multiple credentials are required by the index service to grant a request on behalf of a portal user.



Welcome to the NHSIA Website

Welcome to the NHS Information Authority Portal Site. Use this site to gain access to other NHS Information Authority Websites using the Delivery Areas page or the drop-down menu above.

We aim to be the organisation that the NHS chooses to:

- Support national electronic care records to improve patient care
- Provide information services and knowledge for decision making
- Establish Health Informatics as a recognised national profession
- Provide reliable and secure information infrastructure services

The NHS Information Authority at HC2002 18-20 March, Harrogate

A close look at work leading to the development of a national Electronic Health Record was one of eight sessions run by the NHS Information Authority at 'Healthcare

Fig. 4 *Revealing menu items (see shaded ellipse) based on available privileges*

7. The index server then retrieves the requested client records from its database. Divulging each individual record requires a successful privilege request to be made; that is, the prerequisites of the appropriate OASIS privilege authorisation rule must be met. The index server has its own OASIS policy, used to determine whether records will be released given the credentials of the requester.

Generally, index server policy will require the requester to be in some role in order for the ‘Divulge EHR’ privilege request to be successful. This enables the index to perform a filtering function over the EHR record fragments returned for this request.

Roles that are relevant only at the index server will be defined in the normal manner in the OASIS XML policy file. However, most of the interesting role membership information is actually at the portal. Thus, definitions are included in the index server policy file that mark roles as ‘global’; that is, in this case they will need to have their state retrieved from the OASIS session at the NHS portal.

8. As mentioned in the previous step, the index server will not initially know the roles in which the user is active. This involves gathering required role membership information from the portal site. This process of reverse communication is facilitated using the link token included within the index server session table.

If a large number of records were being searched, performing network communication for each privilege request would rapidly become unnecessarily expensive. One approach is to issue index server role membership based on role memberships at the portal. Of course, this approach requires explicit additions of such role definitions in the index server policy file. For convenience, a credential caching strategy is employed, even in cases where ‘global’ credential checks are being made.

9. The index server’s work in the EHR request is now complete. It returns the filtered list of EHR links to the NHS portal site. Note that at this time there is next to no sensitive information in these EHR records, although future optimisations may store some further information on current medication and allergies for emergency purposes. The index server never provides information beyond that required to contact the Health Care Organisations (HCOs) responsible for each EHR fragment. Indeed in our prototype even this information is proxied through the portal, leaving users unaware from which particular HCO the portal would potentially retrieve this EHR fragment.

Our design originally proposed an index server that merely returned URIs uniquely locating each record, although for convenience a document title field was added in our EHR prototype.

We do not preclude the index server having access to extra databases of policy-relevant material and using these to make its filtering decisions. These databases could facilitate fine-grained policy such as allowing patients control over which individuals have access or are denied access to their individual EHRs. For example, patients might be able to explicitly block certain records from browsing by healthcare professionals, even if those professionals would normally expect to be able to see all the patient’s records. In this manner OASIS environmental predicates could interface with a confidentiality control model such as the TCM discussed in Section 2.3.

10. The portal receives the list of EHR fragments returned by the index server, and formats them for the sake of the

user. In the CBCL OASIS prototype, the records are presented in a numbered list of hyperlinks, each indicating nothing other than their record title. This formatting would probably be sufficient for patient queries, where even in a lifetime the number of EHRs is likely to be manageable on a single long page. For healthcare professionals responsible for generating many patient EHRs per hour for which they will usually remain responsible, more comprehensive filtering and searching facilities would need to be provided.

Each of the hyperlinks to EHRs are actually directed back to the portal, rather than contacting the HCOs directly. Proxying HCO requests through the Portal aims to avoid undue extra resource requirements at each HCO site. For many HCO sites, particularly those using legacy software, it will be more convenient for them to send data in raw, data-oriented messages and rely on the NHS portal sites to perform final data conversion and user-friendly formatting.

We assume that the user selects an EHR from the list presented to them, thus signalling to the portal that the corresponding record should be retrieved.

11. Given that the NHS portal still has the list of EHRs returned by the index service within this session, the portal knows which HCO to contact in order to retrieve the EHR data.

The NHS portal opens a connection to the appropriate HCO Server. Again this connection is an internal OASIS communication, and is done using SOAP messages over an HTTPS channel. Two-sided authentication is performed using the portal and the HCO server certificates.

12. Now the HCO server creates an OASIS session. As for the index server, this session's critical data is stored in a back-end database, and it sets its link field to match the portal's OASIS token value.

The HCO server also has its own independent OASIS policy file. Unlike the portal and the index server, the policy evaluation engine at the HCO server is in a position to make environmental predicate calls whose results are directly related to the contents of the EHR data.

If this HCO site is running a legacy EHR management application, it will be the responsibility of the OASIS-aware software to translate between the Portal's credential structure and privilege requests, and those relevant to the underlying legacy EHR application.

The HCO server is now ready to retrieve details about the requesting user's credentials.

13. The HCO server will make calls back to the NHS portal to retrieve role state. This uses the same mechanism as that for the index server, which was described above.

When the HCO server has all the credential information it needs to perform policy evaluation, it will be able to determine whether or not to grant the requests made by the portal for retrieval of EHR data.

14. The work of the HCO server is now done. If the user's privilege requests were successful, it will retrieve the underlying EHR data. It is assumed that the HCO servers will be able to package their responses into some form of XML communication to be sent using SOAP over HTTPS back to the NHS portal site. In our deployment of CBCL OASIS, we used standard web-based MIME protocols, for example using XHTML to convey formatted text, and PNG or other image formats to provide medical imaging data.

15. Finally, the actual EHR data reaches the NHS portal. If the EHR is delivered in a data-oriented XML message, the NHS portal can apply transformations to make it more readable to the user. In our case XSLT would generally be the most convenient method for formatting XML data, because these services are provided by the J2EE application server environment.

16. At last the EHR data reaches the user's browser. If a multi-window approach was being employed and multiple EHRs had been requested, each EHR will have been laid out in its own block of space (e.g. using tables or frames). In such cases it is assumed that the human user will be able to see correlations between each data block, in preference to trying to program a computer to do so.

Although there are a large number of steps in the above protocol, many parts of the process can be performed in parallel. The abstractions leading to this parallelism are intended to increase the architecture's scalability. At the system level, the J2EE platform provides for server load-balancing; thus more computing infrastructure can be deployed to reduce bottlenecks in the system, particularly replication of the portal, index and HCO site servers.

The CBCL OASIS prototype demonstrates a possible architecture for a distributed, scalable, access control system. It avoids placing undue reliance on the index service for centralised computation, as quickly as possible interacting with the authoritative HCO site servers themselves.

Because the OASIS tokens provide pseudonyms for the duration of a particular session, we can also provide a high degree of privacy protection in the initial stages of communication. It is worth pointing out that the nature of EHRs will eventually make it reasonably obvious who the requests are coming from; however, the Index or an HCO Server will need to know the NHS ID numbers of the records they are managing. Even so, it is possible that a number of role activations need to be performed before privileges are requested. During those role activations it is not necessarily the case that a principal will need to disclose their identity; they may be able to present capabilities issued to them, but which do not encode their personal details.

6 Future work

The CBCL OASIS prototype only scratches the surface of the work required to implement the NHS EHR strategy– there is a great deal of further research and development that needs to be done.

We have focused at a system level on how the policy system could be engineered, and not on how policy administration would occur. In a nationwide deployment, tools would need to be developed to manage versioning and deployment of policy files to the various network components, and to ensure that potential policy conflicts are handled appropriately. It is likely that some degree of database interaction would be required in the access control rule evaluations (for example, to check a principal was not on a blacklist). These environmental predicates need careful attention to ensure that they do not violate the termination requirements of rule evaluation in OASIS.

Further interfaces would need to be designed to allow all users to set the access control policies pertaining to their own records. Many of the other trials of EHR systems have taken a higher-level view of the problem and focused on this user perspective.

At present, the HCOs in our system are simulated. Real HCOs will no doubt uncover further design and engineering challenges, particularly with respect to integrating legacy software into wrappers that can converse with the portal and index servers.

Finally, the system does not currently focus on providing a convenient interface for auditing. At the moment all policy decisions are recorded into the log files at each site, but these files also contain a high volume of other system-related diagnostic information. In a nationwide EHR system, it will be necessary for administrators to be able to investigate particular audit trails, and to merge audit information from different sites. This is particularly relevant to emergency override situations, where the audit logs are the only evidence maintained to check that healthcare professionals do not abuse their privileges.

7 Conclusion

We have presented research leading to the CBCL OASIS electronic health records prototype. Our implementation demonstrates a basic network and role-based security infrastructure on which an NHS EHR system could be built. We have described the different components of our architecture, and provided an example interaction in which a given patient uses a web browser to retrieve securely some of their own EHR records.

The CBCL OASIS EHR project has provided an extremely useful grounding for the OASIS access control architecture through its inclusion in a real distributed system. The NIST RBAC models were found to be insufficient for implementing our prototype. Our use cases required the extra expressiveness of OASIS to support parameterisation, environmental interaction, and appointment.

The CBCL OASIS implementation led to a number of extensions to the original OASIS architecture required to make it work in a web-oriented environment, and to more efficiently handle communication of credentials between OASIS services.

Our EHR prototype has also provided many insights into the security policy management and implementation challenges that still face researchers in the development of a comprehensive nationwide EHR system for the NHS.

8 References

1 Department of Health, NHS Executive: ‘Information for health: an information strategy for the modern NHS 1998–2005’, Jan. 1998, http://www.dh.gov.uk/assetRoot/04/01/43/89/04014389.pdf

2 Hayton, R.: ‘OASIS – an open architecture for secure interworking services’, PhD thesis, University of Cambridge, 1996

3 Yao, W., Moody, K., and Bacon, J.M.: ‘A model of OASIS role-based access control and its support for active security’. Proc. Sixth ACM Symposium on Access Control Models and Technologies, 2001, pp. 171–181

4 Bacon, J., Moody, K., and Yao, W.: ‘A model of OASIS role-based access control and its support for active security’, *ACM Trans. Inf. Syst. Secur.*, 2002, **5**, (4), pp. 492–540

5 Bell, D.E., and La Padula, L.J.: ‘Secure computer systems: Unified exposition and multics interpretation’. Technical report MTR-2997, MITRE Corp., Bedford, MA, July 1975

6 Biba, K.J.: ‘Integrity considerations for secure computer systems’. Technical report TR-3153, MITRE Corp., Bedford, MA, April 1977

7 McLean, J.: ‘A comment on the “basic security theorem” of Bell and LaPadula’, *Inf. Process. Lett.*, 1985, **20**, (2), pp. 67–70

8 Sandhu, R.S., and Samarati, P.: ‘Access control: principles and practice’, *IEEE Commun. Mag.*, 1994, **32**, (9), pp. 40–48

9 Gollmann, D.: ‘Computer security’ (John Wiley & Sons, 1999)

10 Sandhu, R.S., Coyne, E., Feinstein, H.L., and Youman, C.E.: ‘Role-based access control models’, *Computer*, 1996, **29**, (2), pp. 38–47

11 Nyanchama, M., and Osborn, S.L.: ‘The role graph model and conflict of interest’, *ACM Trans. Inf. Syst. Secur.*, 1999, **2**, (1), pp. 3–33

12 Lochowsky, F.H., and Woo, C.C.: ‘Role-based security in database management systems’, in Landwehr, C.E. (Ed.): ‘Database security: status prospects’ (North-Holland Publishing Co., Amsterdam, 1988), pp. 209–222

13 Sandhu, R.S., Ferraiolo, D., and Kuhn, R.: ‘The NIST model for role-based access control: towards a unified standard’. Proc Fifth ACM Workshop on Role-Based Access Control, Berlin, Germany, 2000, pp. 47–63

14 Simon, R.T., and Zurko, M.E.: ‘Separation of duty in role-based environments’. PCSFW: Proc. Tenth Computer Security Foundations Workshop (IEEE Computer Society Press, Silver Spring, USA, 1997)

15 Shin, D., and Ahn, G.-J.: ‘On modeling system centric information for role engineering’. Proc. Eighth ACM Symposium on Access Control Models and Technologies, Como, Italy, 2003

16 Kuhlmann, M., Schimpf, G., and Shohat, D.: ‘Role mining–revealing business roles for security administration using data mining technology’. Proc. Eighth ACM Symposium on Access Control Models and Technologies, Como, Italy, 2003

17 Neumann, G., and Strembeck, M.: ‘A scenario-driven role engineering process for functional RBAC roles’. Proc. Seventh ACM Symposium on Access Control Models and Technologies, Monterey, USA (ACM Press, 2002), pp. 33–42

18 Kern, A., Kuhlmann, M., Schaad, A., and Moffett, J.D.: ‘Observations on the role life-cycle in the context of enterprise security management’. Proc. Seventh ACM Symposium on Access Control Models and Technologies, Monterey, USA (ACM Press, 2002), pp. 43–51

19 Schaad, A., Moffett, J.D., and Jacob, J.: ‘The role-based access control system of a European bank: a case study and discussion’. Proc. Sixth ACM Symposium on Access Control Models and Technologies, Monterey, USA (ACM Press, 2002), pp. 3–9

20 Bacon, J.M., Lloyd, M., and Moody, K.: ‘Translating role-based access control policy within context’, *Lect. Notes Comput. Sci.*, 2001, **1995**, pp. 107–119

21 Becker, M.Y., and Sewell, P.: ‘Cassandra: distributed access control policies with tunable expressiveness’. Proc. Fifth IEEE International Workshop on Policies for Distributed Systems and Networks, Yorktown Heights, USA, June 2004

22 Becker, M.Y., and Sewell, P.: ‘Cassandra: flexible trust management, applied to electronic health records’. Proc. Seventeenth IEEE Computer Security Foundations Workshop, Pacific Grove, USA, June 2004

23 Damianou, N.C., Dulay, N., Lupu, E.C., and Sloman, M.S.: ‘The Ponder policy specification language’, *Lect Notes Comput. Sci.*, 2001, **1995**, pp. 18–38

24 Kagal, L.: ‘Rei: a policy language for the me-centric project’. Technical report HPL-2002-270, Hewlett Packard Laboratories, 4 October 2002

25 Longstaff, J.J., Lockyer, M.A., and Nicholas, J.: ‘The Tees Confidentiality Model: an authorization model for identities and roles’. Proc. Eighth ACM Symposium on Access Control Models and Technologies, Como, Italy, 2003

26 Longstaff, J.J., Lockyer, M.A., and Nicholas, J.: ‘Authorisation models for complex computing applications’. Proc. Information Security Solutions Europe Conf., Vienna, Austria, 7–9 Oct. 2003

27 Longstaff, J.J., Lockyer, M.A., Capper, G., and Thick, M.G.: ‘A model of accountability, confidentiality and override for healthcare and other applications’. Proc. Fifth ACM Workshop on Role-Based Access Control (RBAC ’00), Berlin, Germany, 2000, pp. 71–76

28 Chadwick, D.W., and Otenko, A.: ‘The PERMIS X.509 role based privilege management infrastructure’. Proc. Seventh ACM Symposium on Access Control Models and Technologies, Monterey, USA (ACM Press, 2002), pp. 135–140

29 Chadwick, D.W., and Otenko, A.: ‘RBAC policies in XML for X.509 based privilege management’. Proc. Seventeenth International Conf. on Information Security, Singapore, 9–12 Dec. 2002

30 Chadwick, D.W., and Mundy, D.: ‘Policy based electronic transmission of prescriptions’. Policy 2003: IEEE Fourth International Workshop on Policies for Distributed Systems and Networks, Como, Italy, 2003

31 Bacon, J.M., Moody, K., Chadwick, D.W., and Otenko, A.: ‘Persistent versus dynamic role membership’. Seventeenth IFIP WG3 Annual Working Conf. on Data and Application Security, IFIP, Estes Park, CO, USA, August 2003, pp. 344–357

32 Grimson, W., Berry, D., Grimson, J., Stephens, G., Felton, E., Given, P., and O’Moore, R.: ‘Technical description: Federated healthcare record server – the Synapses paradigm’, 1996, http://www.cs.tcd.ie/synapses/public/html/technicaldescription.html

33 Baker, D.B., Barnhart, R.M., and Buss, T.T.: ‘PCASSO: applying and extending state-of-the-art security in the healthcare domain’. Proc. Thirteenth Annual Computer Security Applications Conf. (IEEE Computer Society, Silver Spring, USA, 1997), p. 251

34 Bacon, J.M., Moody, K., and Yao, W.: ‘Access control and trust in the use of widely distributed services’, *Lect Notes Comput. Sci.*, 2001, **2218**, pp. 295–310

35 Hine, J.H., Yao, W., Bacon, J.M., and Moody, K.: ‘An architecture for distributed OASIS services’, 2000, **1795**, pp. 104–120

36 W3C: SOAP version 1.2 part 0: Primer, http://www.w3.org/TR/soap12-part0/, June 2003