Implementing Role Based Access Control for Federated Information Systems on the Web

Kerry Taylor and James Murty
CSIRO Mathematical and Information Sciences
and CRC for Enterprise Distributed Systems Technology
GPO Box 664, Canberra, ACT 2601.

Kerry.Taylor@csiro.au
James.Murty@csiro.au

Abstract

There is rapidly increasing interest in Australia in on-line sharing of information stored in corporate databases, especially within and between staff of independent government agencies. Biological collections databases and population health GIS are good examples of the frequent situation where database custodians are looking for dynamic, distributed, heterogenous federated information system models for information sharing within loosely constituted communities. This paper describes a security model for authentication and access control to federated systems. The model supports single sign-on for users; a high level of autonomy for database custodians; and a low maintenance overhead. The model's implementation using PKI and Web technology is described.

Keywords: RBAC, Federated Databases

1 Introduction

There is widespread recognition of the value of sharing access to data held in diverse, distributed databases amongst professional communities. Data resources of interest in a given context are often maintained by independent authorities with loose or non-existent previous relationships, such as separate business units of a large company or different local, state and federal government agencies. In such cases great value can be gained by offering integrated access to all these data sources through a single mediated point, significantly simplifying the data query process and enabling the retrieval of much richer and better presented information than is possible from individual data sources. However, there is widespread concern about maintaining data privacy in the case of data that is subject to commercial, scientific or personal confidentiality and fear that this data could be misused, intentionally or otherwise, to the detriment of financial, environmental or social goals. Whenever data distribution is achieved through person-to-person interaction these concerns can be addressed on an as-needs basis. But when data is distributed electronically over communications networks, at the instigation of the data receptor rather than the data provider, careful attention needs to be given to formalise appropriate principles of data access and to implement those principles in data access software.

Decisions about who should access what data are usually best addressed by the usual custodians of the data: those most responsible for capturing, maintaining and interpreting it. This paper discusses the issues in custodial maintenance of data access in a context of large-scale Web-based integrated information systems and proposes an administrative and technical framework that addresses those needs. The framework suggested is a general one, permitting variation according to domain-specific security requirements. To demonstrate this we discuss its implementation in two distinct distributed systems built using the Internet Marketplaces architecture and toolkit (IMP) developed at CSIRO (Abel *et al* 1999). The systems are Bioplex, aimed at a scientific community for sharing information in biological specimen databases, and the New South Wales Health GIS pilot (PHIMP), aimed at sharing population-scale health indicators within a large and distributed health authority.

This paper is concerned with securing read-only access to sensitive data as it is transmitted and delivered as part of federated database projects: it does not address issues relating to access to the data by other means (for example, directly to databases via internal networks), nor opportunities for potential security breaches into other networking resources via the IMP systems. While these matters are important and are addressed by the design, they are outside the scope of this paper.

We begin with a brief outline of the system architecture into which the access control model is incorporated. In the following section we discuss some of the access control requirements evident in loosely-coupled federated information system communities. In the next section we briefly introduce the technological concepts and tools that we leverage in developing secure access control mechanisms to address the requirements. We then discuss the security model itself describing how we achieve client

The work reported in this paper has been funded in part by the Co-operative Research Centre for Enterprise Distributed Systems Technology (DSTC) through the Australian Federal Government's CRC Programme (Department of Industry, Science & Resources).

authentication and custodian-specified access control. Finally we describe two distinct implementations of the security model which demonstrate its flexibility. We conclude with a summary of the features and limitations of the model.

As illustrated in figure 1, IMP systems are based on a three-tiered architecture comprising a client level, a broker level, and a gateway level. Users access the system using light-weight client software such as a standard Web browser or via Java applets run within a browser. They interact with a Web site offered by a broker that offers an integrated interface to the underlying, searchable local databases. The databases themselves remain *in-situ* at the third tier and are made accessible to the broker via the Web with the addition of some gateway software at the database site and standard Web protocols (HTTP/HTTPS/SOAP) for information exchange. Communication between a broker and the gateways is coarse-grained and stateless. The architecture permits a single database to offer multiple interfaces through its gateway to multiple brokers: each broker may offer specialised services appropriate to a target user community. From the gateway perspective a target user community may be identified by a *profile* which corresponds to some exposed representation of the underlying database, where this representation is of a scale appropriate for modelling the information requirements of each federated system in which the database participates. For example, a profile may be mapped to a schema, a GIS layer, a file, or a database table. A single gateway may make its underlying data available through multiple profiles, allowing a data provider to participate in multiple user communities while minimising the administrative and configuration overhead of this participation.

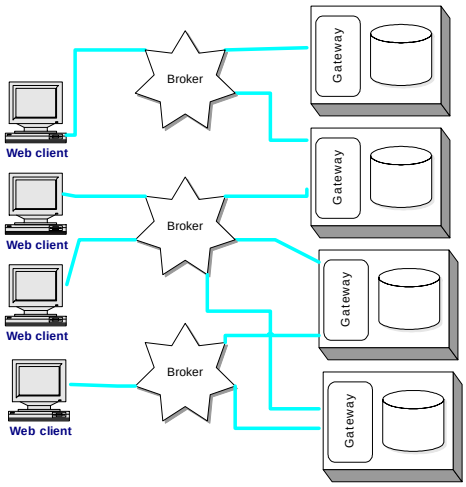

Figure 1: Internet Marketplaces Architecture

2 Access Control Requirements

In read-only federated information systems, any security model must support the following forms of access control (in order from coarse- to fine-grained control):

1. Minimum user identification requirements whereby users with inadequate credentials are not permitted any form of access to the custodian's system. Example credential requirements are that users access the system from a trusted network domain, provide username/password information, supply an X.509 or Kerberos personal certificate, etc;
2. Profile restrictions which permit access to particular data classes (or views) exposed by the gateway;
3. Record-level content restrictions which may suppress entire records from the result dataset provided to a user with limited access privileges;
4. Attribute-level content restrictions which may suppress sensitive attribute fields from the results returned to a user's query, and finally;
5. Attribute obfuscation, whereby sensitive attribute data is dynamically modified in the result set in such a manner that renders it suitable to be provided. Examples of obfuscation might include the modification of address data to remove street names and numbers, or of spatial point data to reduce co-ordinate precision. This simple technique is commonly used for privacy protection in systems of medical records (Sweeney, 1998).

There is wide recognition of the need for access capability to depend on the *role* of the accessing user within a system (Sandhu *et al* 1996). Users may be partitioned into roles (also called access classes) that govern their access to the underlying database. In the biological collections domain, for example, such roles might include scientific collector, fellow employee, commercial partner, quarantine officer, or member of the general public.

It seems quite feasible to require communities of interest to come to a level of agreement on what kind of information (profiles) should be available to users in the community, and to be able to categorise these profiles according to their ability to meet the needs of classes of users within the community. For example, a biological collection community may determine that a profile designed for scientific collectors should include the *collector* attributes (identifying the name of a specimen collector because that gives some insight into the value of the specimen), but not the *location* attributes, identifying the place at which a specimen was found (because this may be of scientific competitive advantage). On the other hand, a profile for a quarantine officer may include *location* (because a specimen collected at an international airport has a very different quarantine status to one collected in an agricultural district) but exclude *collector* (because specimens may have been collected illegally). Where such agreement is reached, the data providers participating in a community can use the shared concepts to inform the implementation of role-based access restrictions at their gateway access point. The federated system developers can use the shared concepts (named as a profile) to develop the information model to be offered to the end users.

Although such communities can agree on the information content to be presented, it seems much more difficult to develop a uniform classification of an individual user, or even a group of users that are associated by a common

affiliation. While a user might be considered of low security risk by one database custodian (such as when that user is a member of the same corporate group or institution as the provider) another provider might well regard the same user as an unknown member of the general public. This creates a clear requirement for access rights to be directly controllable by the custodians of individual databases. That is, although the classifications of roles and their access permissions might be agreed in data custodian communities, the rules that govern participation of a specific user in a role must be locally controlled. This contrasts with the model proposed by Park *et al* (2001), where user-to-role mapping is the responsibility of a centralised role broker. Furthermore, it prohibits a mandatory access control policy based on multilevel security classes as centralised control is unsatisfactory. The rationale for data source- (or server-) based control of user-to-role mappings is discussed in depth in Bull *et al* (1992).

Different security measures are appropriate for different roles. For example, a user in a "general public" role could be heavily restricted in terms of access to content, but should not require any sign-on process, encryption facility, or pre-arrangement for access. By contrast, a user in a more privileged role could have access to much more data but would need to be individually authorised in advance by each custodian, and may be required to sign up to formal condition-of-use agreements. An access role in our model is a logical grouping of users applicable to one or more profiles. It is likely that when a gateway participates in multiple user communities, and thus offers multiple profiles of its data, these communities will have some roles in common, reducing administrative overhead. For example, the "general public" role definition will be reusable for any profile for which security protection is not required.

When assigning access rights to roles, we should consider whether the model should mandate linear ordering of roles with respect to increasing access rights. In this case we could admit a user to a single role (per site) and agree that admission implies admission to all less-restricted roles classes at that site. This would have advantages in ease of administration (since only a single choice needs to be made per user), but limits flexibility if classes do not fall naturally into an ordered structure. A linear order would not permit, for example, one user to be allowed to access attributes about collector but never about location, and another to view location but never collector. Although hierarchical structures for access rights can deal with this option (Sandhu *et al* 1996), it requires a rather large number of distinct roles in the domain in order to benefit from the added complexity introduced. Therefore, we do not advocate role ordering, but instead require that roles are explicitly allocated to any number of profiles, and that users may access any of those profiles for which they are role members. In environments where complicated role relationships must be captured, management techniques such as those proposed by Nyanchama and Osborn (1994) may be employed.

Recalling that the individually administered databases are integrated into a coherent view, from the user's perspective, it is highly desirable that a user undergoes a single sign-on process for access to all the databases they may query through that view. This implies that there must be a role for the broker in centralising the sign-on process, while delegating authority for access control to the local database custodians. The custodians need identifying details about the user in order to do this. In the proposed model a broker is required to collect client credentials at sign-on and relay this information to the distributed databases with each query request.

Furthermore, we propose that requests for data via the federation explicitly name (one or more) profiles as a context for the request. Access decisions can be made at the profile level without finer-grained analysis of the query itself (provided the query answering method ensures that only queries that are relevant to a profile are answered). A profile corresponds to a *permission* in the role-based access control literature (Sandhu *et al* 1996), although it refers only to the permission to read any of the data represented by the profile.

Finally, there are some operational requirements on any implementation of these principles. In particular, administration tasks such as admission to and removal from classes must be low-cost, especially for the less sensitive classes which might be expected to have larger user memberships. Although the low-cost argument applies to the individual databases, it applies even more sharply to administration required at the broker, for which, in the worst case, administration tasks must be performed for each element of the cross-product of users and databases.

3 Security Infrastructure

This section introduces the basic concepts and terminology of secure Web communications as a precursor to discussion of our solution. The explanation given here is considerably simplified: a fuller explanation is given in (Hirsch 1997).

The Secure Socket Layer (SSL) protocol, an open (de facto) standard designed by Netscape is the most widely accepted basis for secure Web communications, and is the foundation for the secure version of the HTTP Web protocol known as HTTPS. SSL uses public-key cryptography, based on the exchange of standard X.509 certificates for both authentication of communicating partners and encryption of data in transit. SSL provides *authentication*, that is identification of communicating partners; *privacy*, that is protection of communicated information from interception by third parties, and *integrity*, that is the prevention of interference with intended message content. SSL support is built in to most Web browsers and servers, enabling near-transparent use once configured.

Any PKI-based security system relies for authentication on the existence of one or more trusted certificates (CA certificates) which may be used to *sign* other certificates, indicating that the owner of the trusted certificate vouches that the owner of the signed certificate is indeed who they claim to be. An entity that trusts the owner of a given certificate is also supposed to trust those certificates

signed by this trusted owner. Thus the set of entities trusted in the system forms a hierarchy, with the explicitly trusted certificates at the top.

4 IMP Security Model

There are two levels of security in the IMP security model, network transport and access control. At the network transport level there may be a requirement that communication channels be initiated only by properly authorised parties in the system, and that the data transferred between these parties be protected in transit. Such requirements can be met using commonly available HTTPS/SSL infrastructure as described above. Access control may be required in addition to any transport level security provisions and is implemented in the gateway software. The gateway is responsible for implementing the access restrictions outlined above, according to parameters configured by the database custodian. A gateway may also be configured to ensure that the network transport through which it is accessed uses sufficiently secure protocols.

4.1 Network Transport

4.1.1 Client Authentication

Let us first examine how user authentication and secure data transmission operates between the broker and the Web clients. The broker acts as a single sign-on point for clients and is therefore responsible for ensuring that any client information required by the downstream gateways is collected and relayed to each gateway with a client's request. The client information required by an IMP system (and thus by the broker) can vary considerably depending on the specific security requirements of the environment in which the system is running and the nature of the databases exposed. For example a client's membership of a particular network domain may suffice for identification purposes in an intranet-deployed system, while stronger mechanisms such as passwords or client certificates may be necessary when the system is available via the internet. These decisions are implemented through run-time configuration parameters in the software.

The use of IP/DNS machine addresses or username/password challenges for client authentication is commonplace: facilities for doing this are available in all standard web servers. In cases where this kind of authentication suffices the only administrative burden placed on the broker's managers is maintenance of a database of accepted machine addresses or username/password pairs. In more demanding environments, where client certificates (X.509 or Kerberos) are required, a certificate hierarchy must be managed.

In our model, we recommend that a simplified hierarchy be implemented where the only trusted certificate is that owned by the broker, and only those certificates signed by the broker's certificate directly are considered valid (i.e. the depth of the signing hierarchy is no more than 1). This approach reduces administrative complexity by allowing verification of a client's identity at one point only, the broker. It also increases security as only those clients verified directly by the broker's managers may access the system: a signed client may not vouch for the identity of a client unknown to these managers.

Using this approach a client gains access to an IMP system requiring certificate identification by applying for a certificate from the managers of the broker, as trusted representatives of the information system community. Included with this application, electronic or otherwise, must be sufficient information for the managers to adequately verify the applicant's identity. Assuming all is well a certificate containing information about the client, and signed by the broker CA certificate, will be provided. The client will then supply their personal certificate when querying the broker to gain access to the system, and the information contained in their certificate will be relayed by the broker to each gateway from which it draws information. The broker's managers, acting as a certificate authority, must be assured that the information about a user that is written into a certificate is correct, because this information will be relied on for access control throughout the system. Database custodians will rely on the broker's managers to authenticate users correctly.

The major advantage of using certificates instead of mechanisms involving a database of machine addresses or passwords is that the broker need not maintain a persistent record of users. Once the client's certificate has been generated, a once-off process, all the information necessary for system access is encoded in the certificate itself which is held by the client. This means there are no on-going administrative requirements to manage updating of passwords at the broker or gateways, nor is there the risk that the credentials of all the system's users will be available should the central database be compromised. Additionally, client-identifying information used by the gateways to make access control decisions is only ever transmitted over encrypted, secure SSL channels, making spoofing attacks impossible. Such security cannot be guaranteed by mechanisms that do not mandate data encryption.

4.1.2 System Component Authentication

In deployment of the three-tier architecture, communication channels between brokers and gateways also need to be secured.

We recommend that SSL be ordinarily used between these components. This may be unnecessary in cases where the network on which the system resides is completely protected by other means. But this situation is rare in loosely-coupled federated information system communities, and requires a large degree of trust in correct network configuration. Network configuration may be beyond the control of those maintaining the components (that are typically embedded within business units of an organisation rather than under corporate IT management).

Based on the certificate hierarchy detailed above, which allows verification of Web client identity, it is

straightforward to extend the hierarchy to facilitate secure broker-gateway communication. Each gateway may be considered a client of the broker and be required to identify itself with a signed certificate. Conversely, the broker will identify itself by providing its own certificate to each gateway it contacts to fulfil a query. Thus every entity in the system must identify itself with a certificate signed by the broker's certificate, or the broker certificate itself in broker-to-gateway identification. This means that the broker can be sure it is communicating only with authorised gateways when sourcing data, and that each gateway can be sure it is providing data only to a valid broker.

Such an arrangement can be used between back-end components of a system regardless of whether client certificates are required from the end-user Web clients or simpler authentication mechanisms are deployed. Maintenance is low, and is limited to the re-generation and installation of certificates when the broker's certificate expires or the broker is moved to a new host platform. The added security assurance is transparent to end-users.

4.2 Access Control

We propose a two-tiered mechanism for authentication and access control that has the broker perform user authentication but leaves access control decisions to the database custodian at the gateway site. The broker authenticates Web clients using one of the mechanisms discussed above and includes the client information with each query request directed to a gateway. The gateway software is aware of access control configuration settings specified by the gateway administrator, generally in a text document. The process for evaluating - the client's request proceeds as follows. These steps correspond roughly to the access control levels suggested previously. If a client's request fails to pass any of these checks the query is aborted and an error message returned to the broker.

1. When a connection is made (from the broker to a gateway) the gateway software first checks that there is sufficient client information provided for it to make an informed access decision. For example, a gateway may require that a client's username and password be provided, or that a client be identified with an X.509 certificate;
2. The gateway uses rules specified in the access control configuration to map a client into zero or more roles based on the client's information. This information can include machine address, username/password, or X.509 certificate fields. These fields include name, organisation, organisational unit, locality, state, country, email address and validity period or community-agreed additional information (CCIT 1988). The client must be assigned to at least one role, and from this point on access control decisions are based on role membership;
3. The client's query itself is then examined to determine which data profiles the query references. The gateway checks that each of these is accessible to at least one of the roles of which the client is a member. Note that this means that a user does not explicitly associate a role with a query; *any* suitable role for which they are authorised will do.
4. The allowed query is then processed in a manner appropriate to the data source. This processing is aware of the role applicable to the request, in which case data may be selectively suppressed (at the record or attribute level) or modified (at the attribute level) to provide only that information (or precision of information) appropriate for retrieval by the client.

The mechanisms for performing selective suppression and obfuscation (step 4) will depend on the data source exposed by the gateway and may require database-specific customisation and the addition of information to the data set, such as a flag indicating the access level required to view a particular record or attribute. The previous three steps are implemented in the generic gateway software.

Access control is thus governed by software at each gateway site and configured according to the requirements of the database custodian. The custodian is responsible for determining roles, determining membership of those roles, and determining the data retrieval permissions applicable to those roles.

Below is an example access control configuration document. The IMP toolkit supports the specification and interpretation of this XML document format for deployment in federated systems. Information irrelevant to the examples has been removed and the interesting paragraphs have been numbered and highlighted for reference in the following explanation. The underlying role mapping language is based on that employed in the Apache web server for X.509-based access control (Engelschall, 1999).

```
<ImpSecurity>
1:    <User2Role roleName="publicAccess">
            <AnonymousConstraint>
                  <IPAddress>*</IPAddress>
            </AnonymousConstraint>
      </User2Role>

2a:   <User2Role roleName="HRdepartment">
            <X509Constraint>
                  <OrganisationalUnit>Human Resources</OrganisationalUnit>
                  <Organisation>BigOrg</Organisation>
            </X509Constraint>
      </User2Role>

2b:   <User2Role roleName="HRdepartment">
            <X509Constraint>
                  <CommonName>John Smith</CommonName>
                  <OrganisationalUnit>Executive</OrganisationalUnit>
                  <Organisation>BigOrg</Organisation>
            </X509Constraint>
      </User2Role>

2c:   <User2Role roleName="HRdepartment">
            <AnonymousConstraint>
                  <DNSAddress>*.accounts.bigorg.com</DNSAddress>
            </AnonymousConstraint>
            <BasicConstraint>
                  <UserName>auditor</UserName>
            </BasicConstraint>
      </User2Role>

3:    <Role2Profile roleName="publicAccess">
            <Profile>Public</Profile>
      </Role2Profile>

4:    <Role2Profile roleName="HRdepartment">
            <Profile>Confidential</Profile>
      </Role2Profile>
</ImpSecurity>
```

In the example above there are two roles, *publicAccess* and *HRdepartment*. Any user whose IP address matches the '*' wildcard (every user) is assigned to the *publicAccess* role (1). Members of this role are explicitly allowed to perform queries on the *Public* profile (3). Any user who provides a certificate identifying them as a member of the Human Resources department in the BigOrg company is assigned to the *HRdepartment* role (2a). John Smith, a member of the executive of BigOrg, is also assigned to this role when he supplies his personal X.509 certificate (2b). Finally, any user who provides the username 'auditor' from a computer in the 'accounts' domain of BigOrg's network is also assigned to the *HRdepartment* role (2c). All the *HRdepartment* role members are allowed to access data in the *Confidential* profile (4). In this example members of the *HRdepartment* role will also be assigned to the *publicAccess* role, and are therefore granted access to both of the available profiles. The contents of each profile, as mapped from the underlying data source, is configured in separate schema document: discussion of that document is beyond the scope of this paper.

The user credentials used in this example, X.509 certificates and usernames, are standard authentication mechanisms and are automatically checked by the server engine running the IMP service. Thus only valid X.509 or username credentials are used for evaluation of a user's access rights.

5 IMP Secure System Implementations

We discuss below the implementation of access control in two IMP systems based upon the proposed model. Specifically we describe the security mechanisms that we employed in response to the different distribution environments encountered, and discuss the relative advantages and disadvantages of each approach.

5.1 Bioplex

The Bioplex, illustrated in Figure 2, (Leow & Taylor 2000), offers access to heterogeneous biological collection databases via the internet to both trusted parties (such as researchers, collectors) and the general public. Some of the information made available has high potential for misinterpretation or could be used to harm Australia's trade interests. It was therefore vital that access to potentially sensitive data sets and data elements be limited to authorised persons, and that this data be encrypted for transmission over the internet. At the same time the non-sensitive data had to be accessible to anyone.

To fulfil these objectives we mandated a strong PKI infrastructure where all entities in the system, including clients, are required to identify themselves with X.509 certificates and communicate using the SSL/HTTPS protocol. At the data source (gateway) level access control decisions are based on client information contained in these certificates. Where a database in the system contains sensitive record or attribute values these values may be selectively suppressed based on the role membership of the client. This suppression requires database-specific modification of the gateway software.

This approach provides a system meeting the security requirements of this particularly strict domain. Unfortunately the requirement that clients supply X.509 certificate identification makes initial use of the system difficult as each client must obtain a certificate. This problem can be avoided by providing a separate broker entry point for non-authenticating users, where a "General Public" client certificate is forwarded to gateways on the client's behalf. Clients (or groups of clients) who wish to access sensitive data are required to apply for their own unique certificate. Having obtained this from the broker administrators they must make contact with each gateway for which they wish to arrange special access conditions, which may be granted to them by the gateway administrator once the necessary condition-of-use arrangements have been agreed. Gateway administrators therefore have a high degree of control over who may access sensitive data.

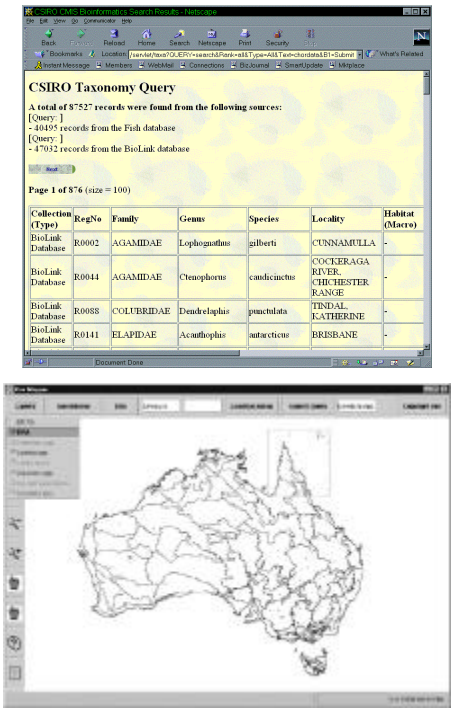

Figure 2: Bioplex system

5.2 New South Wales Health GIS Pilot

The New South Wales Health GIS (Geographic Information System) Pilot (PHIMP) is a project instigated by the NSW Department of Health to provide web-based access to geographically-based health information to employees. Information is sourced from spatial databases in departmental offices distributed throughout the state of New South Wales. Clients access the PHIMP broker via the department's secure intranet, and gateways in the system are generally accessible via this intranet. Profiles are mapped to GIS layers: some of them are sensitive even within the intranet environment. Therefore gateways must be protected from access by entities other than the user-authenticating broker.

A simplified PKI infrastructure was employed for this system in which broker-gateway communications use the SSL/HTTPS protocol and require X.509 certificate, while clients are identified by a username and password only. When clients log in to the broker they are prompted for their credentials using the standard HTTP Authentication mechanisms available in any web browser. At the gateways, access control decisions are made based only on clients' usernames.

This approach allows for strong security between the broker and gateways in case communication occurs over unsecured networks, while making the client login process simpler than is the case in the Bioplex system. By using employees' pre-existing usernames as identifiers, the broker's task of authenticating clients could be performed using existing enterprise employee authentication systems, such as a database or LDAP server. This effectively leverages the enterprise's existing IT infrastructure.

6 Features and Limitations of the Model

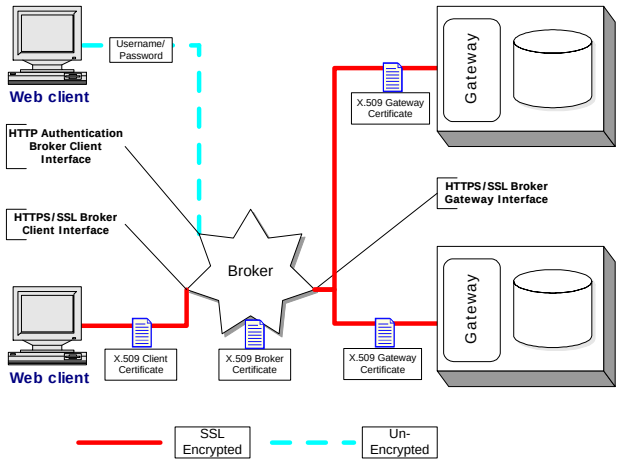

Figure 3. Internet Marketplaces Security Architecture

Figure 3 summarises the security model. The model does not rely on centralised agreement on the roles of users—neither the system of roles (their names and relationships) nor the membership of users in each role. Instead, individual database custodians can determine which users have access to which data, on an individual basis or according to institutional membership or other user features. The cost of procedures for granting data access to users can be commensurate with the sensitivity of the

data, because the rule language permits custodians to rely on any or all of a wide range of user descriptors available through the model.

Security measures based on strong encryption and reliable standards are available to protect data on communication links and at network interfaces, ensuring authentication, privacy and integrity where these communication properties are required. In systems with less rigorous security requirements other standard identification mechanisms, such as machine address or username/password challenges, may be used.

Once a user has registered for access to the broker the system offers single sign-on per session. The user may then access all the database services for which they have been given permission by local custodians. A user granted access to sensitive data would normally also have access to other presentations of the less sensitive parts of the data, subject to the formulation of the mapping rules by local custodians. The model is dependent on a workable system of trust within loosely federated communities: custodians must trust any brokers they authorise to both authenticate users and deal appropriately with information transmitted to them on behalf of users. But custodians entirely retain control over the allocation of access rights to users through the role and profile modelling mechanisms.

7 Related Work

The approach described here implements the $RBAC_0$ model in Sandhu *et al*'s (1996) taxonomy of role based access control models: it includes users, roles, permissions, sessions and explicit mappings between them. It deliberately avoids role hierarchies ($RBAC_1$ and $RBAC_3$) in order to simplify the implementation of mappings for database custodians, and avoid the constraints ($RBAC_2$ and $RBAC_3$) as generally unnecessary in the domain of application (read-only federated databases). As a slight variation to the $RBAC_0$ model, we do not require consistency of role labels across the system, as multiple use of role labels across local databases has no effect on the global behaviour (although unique names could be trivially created by a prefixing scheme if desired). Role labels *could* be standardised within a community, which may offer some advantage in human communication and administration, but this has no effect on the technical implementation of the model described here.

Jonscher and Dittrich (1994) introduced the concept of authorisation autonomy to federated database design: whereby database custodians control which global users can access their data—also an important consideration in this paper. They suggest three alternative authentication schemes: the first is claimed to be best for autonomy but is cumbersome because it requires multiple password entry and administration (a problem we have specifically set out to avoid). The second is very much like what we propose: the broker is responsible for validating the identity of users and for forwarding the identification to the local user. In the third scheme the local database custodians cede authorisation control to the broker and cannot implement their own access control. Our model can also support the third scheme when custodians chose to ignore identity information available to them. In addition, we provide mechanisms in the local database wrapper whereby varying combinations of user attributes such as client IP address, organisation, and full names can be used to implement tighter authentication requirements for more sensitive data. A disadvantage of the method is a relatively high administration cost: a user can access some data only if authorisation is granted by both the global administrator and the local custodian.

In another federated database system IRO-DB (Essmayr 1996), an approach was developed for access control in federated and o-o databases. It differs from the work in this paper by dealing comprehensively with modelling information objects in a class hierarchy and also modelling authorisation types (read, write, delete, etc) in a hierarchy, and then managing the conflicts in positive and negative authorisation rules expressed in those terms. In IRO-DB all authorisation and access control is managed and enforced centrally, at the federated level.

De Capitani di Vimercati and Samarati (1996) propose a model for federated databases that, more flexibly than ours, requires user authentication at the global level but also permits additional user authentication at each local site if required by the local custodian. Like ours, their model does not assume particular data models are used in participating systems. However, their model assumes a federated administrator responsible for maintaining authorisations on global objects, and it requires *all* local databases involved in responding to a global request to authorise their part of the request in order for any answer to be returned via the federation.

None of these works address network layer security nor implementation within a coarse-grained Web services-based architecture.

Recently, there has been considerable activity in the development of standards in the area of Web system security. There is a proposal for NIST (csrc.nist.gov) standardisation of an RBAC reference model (Ferraiolo *et al* 2001), together with requirements for system administrative functions and access control. The OASIS consortium (www.oasis-open.org) has just ratified SAML (security assertions markup language) for exchanging information between sites about authentication, attributes and authorization, to support single sign-on in federated systems. SAML authentication and attribute statements have direct application in our model as they include the information being transferred between components in our approach. SAML authorisation statements could also be applied if we wish to communicate access control decisions from local sites back to the federation, independently of responses to data requests.

8 Conclusion

The proposed model focuses on the need to have access control decisions under the control of distributed database custodians. This is achieved by a separation of concerns that has user authentication performed once for each user by the central broker, thus minimising administrative

overhead, but retaining access control for authenticated users in the hands of database custodians. The role based access control model proposed is the simplest of standard reference models (Sandhu *et al* 1996) but is sufficiently flexible for the intended purpose and easier to administer than enhanced models. The model builds on the basic tools for secure Web communication, augmenting them with application software at the broker and at database gateway sites. A feature of the proposal is that it permits access control lists to be managed such that different levels of identification can be used to enable access to profiles according to the sensitivity of information represented by the profile. We have configured the model to apply to two federated database applications with different security needs.

Acknowledgement

Assistance in the early stages of this work was given by Ming Yung, Matthew Wilson, Jian Yang, Drew Devereux, Martin Kuchlmayr and Mike Kearney. Tim Churches and Alan Willmore collaborated on the PHIMP project. Patrick Hung helped to improve the paper's presentation, and the anonymous referees suggested many valuable improvements.

9 References

Abel, D. J., Gaede, V. J., Taylor, K. L., and Zhou, X. (1999). SMART: Towards spatial Internet marketplaces. *Geoinformatica*, **3**(2):141-163. June 1999.

Bull, J A., Gong, L., Sollins, K. (1992) Towards Security in an Open Systems Federation, *Proceedings of European Symposium on Research in Computer Security*, Springer LNCS 648 pp 3 – 20.

CCITT 1988. *X.509: The Directory – Authentication Framework* CCITT Blue Book, Vol VIII pp 48-81. CCITT 1988.

De Capitani di Vimercati,, S., Samarati, P. (1996): Access control in federated systems, *Proceedings of the 1996 workshop on New Security Paradigms*, pp 87--99, September 17-20, 1996, Lake Arrowhead, California, United States.

Engelschall, R. S. (1999), mod_ssl 2.4 User Manual, The Apache Interface to OpenSSL, 1999. www.modssl.org.

Essmayr, W., Kastner, F., Pernul, G., Preishuberand S., and Tjoa, A. (1996): Authorization and Access Control in IRO-DB, in Stanley Y. W. Su (ed), *Proceedings of the Twelfth International Conference on Data Engineering*, February 26 - March 1, 1996, New Orleans, Louisiana. pp 40—47. IEEE Computer Society.

Ferraiolo, D., Sandhu, R., Gavrila, S., Kuhn, D. and Chandramouli, C. (2001): Proposed NIST Standard for Role-Based Access Control, ACM Transactions on Information and System Security, **4**(3):224-274.

Hirsch, F. J. (1997), Introducing SSL and Certificates using SSLeay, *World Wide Web Journal*. **2**(3) Summer 1997. www.ultranet.com/~fhirsch/Papers/wwwj/index.html.

Jonscher, Dirk and Dittrich, Klaus R. (1994): *An Approach for Building Secure Database Federations* in Jorge B. Bocca and Matthias Jarke and Carlo Zaniolo (eds), VLDB'94, *Proceedings of 20th International Conference on Very Large Data Bases*, September 12-15, 1994, Santiago de Chile, Chile. pp 24—35. Morgan Kaufmann.

Leow, R. and Taylor, K. (2000), Efficient Web Access to Distributed Biological Collections Using a Taxonomy Browser, in O. Gunther and H.-J. Lenz (eds), *12th International Conference on Scientific and Statistical Database Management*, Berlin, Germany, July 2000. pp 25—38. IEEE Computer Society.

Nyanchama, M., Osborn, S. (1994), *Access Rights Administration in Role-Based Security Systems*, IFIP WG 11.3 Database Security, 1994.

Park, J., Sandhu R., and Ahn, G.-J. (2001), Role-Based Access Control on the Web, *ACM Transactions on Information and Systems Security*, **4**(1). February 2001.

Sandhu, R, Coyne, E., Feinstein, H., and Youman, C. (1996): Role Based Access Control Models, *IEEE Computer* **29**(2), February 1996.

Seeney, Latanya (1998), Three Computational Systems for Disclosing Medical Data in the Year 1999, in Cesnik and McCray (eds), *MEDINFO 98*, IOS Press Amsterdam.