Medical Informatics Europe '97
C. Pappas et al. (Eds.)
IOS Press, 1997

Security of Healthcare Information Systems Based on the CORBA Middleware

Bernd BLOBEL and Martin HOLENA
Otto-von-Guericke University Magdeburg, Faculty of Medicine, Institute of Biometrics and Medical Informatics, Leipziger Straße 44, D-39120 Magdeburg, Germany

Abstract. The development of healthcare systems in accordance to the "Shared Care" paradigm results in co-operative health information systems across the boundaries of organisational, technological, and policy domains. Increasingly, these distributed and heterogeneous systems are based on middleware approaches, such as CORBA. Regarding the sensitivity of personal and medical data, such open, distributed, and heterogeneous health information systems demand a high level of data protection and data security, both with respect to patient information and with respect to users. The security concepts and measures available and additionally needed in health information systems based on CORBA architecture are described in this paper. The proposed security solution is also open to other middleware approaches, such as DHE or HL7.

1. Introduction

Under the constraints of changed basic conditions of care and of increasing demands for health services, all developed countries are modifying their healthcare systems structure to a shared caring concept, extended communication and co-operation between direct as well as indirect care providers, and at least a minimum of competitiveness.

These objectives must be supported by adequate, i.e. distributed and really co-operative health information systems. Meeting these requirements, the newly developed or legacy components of information systems must realise Leguit's integration type "Integration" [3]. Only strictly object-oriented approaches, providing services comprising data *and* methods applicable to the data, are capable of such co-operation. Regarding domain specific affiliation, the achieved integration level, and platform or provider dependencies, different approaches to the middleware layer below end-user applications attempt to tackle the isolation and not-openness of legacy health information systems. Including the healthcare domain specific vertical common facilities envisaged by the CORBAmed group, the present paper is restricted to CORBA (Common Object Request Broker Architecture) as the approach probably best fulfilling the above requirements. However, the proposed security solution is also applicable to the other middleware approaches using CORBA services. A comparative study of middleware approaches for healthcare (HL7, DHE, CORBA) can be found in [1].

2. Overview of the CORBA Middleware Approach

Based on the popular object-oriented paradigm, the CORBA approach is being elaborated by the Object Management Group (OMG), created in 1989 to promote the theory and practice of object technology in distributed computing systems. An object, in general, can represent anything that is unambiguously identifiable and capable to provide one or more services

that can be requested by some kind of client. Associated with each object is a set of methods and a set of attributes. The former represent the provided services, the latter represent the state of the object and the information passed during the request or produced when services are provided. Each implementation of an object consists of three parts:

- Operations, implementing the services represented by the object's methods;
- Data, which implement the object state and information represented by the attributes;
- Interface, implementing the ability to accept requests and to return information. It is specified by a special Interface Definition Language (IDL) defining passed parameters, return mode, as well as links to a request context and to exception handling methods.

The basic structure of the CORBA architecture is the Object Request Broker (ORB), which is responsible for locating an object implementation, preparing it to receive the request and communicating the data making up the request. Hence, it plays the role of an active object interconnection bus. It consists of the ORB core, which provides the basic representation of objects and communication of requests, and various client- and/or server-side interfaces. Though the ORB can be implemented in various ways, its interfaces are standardised and implementation-independent. They provide CORBA with a high degree of portability, scalability and flexibility.

Between the ORB and the application objects, implementing end-user applications, the CORBA architecture situates two layers of widely used objects. The lower layer of Common Object Services provides basic functionality for using and maintaining objects. Examples of common object services include Transaction Services, Object Lifecycle Management, Event Notification, or Concurrency Control. The higher level of Common Facilities provides general purpose capabilities useful in many applications. The facilities may be horizontal, pertaining to different application areas, or vertical, within a particular application area like healthcare. Available horizontal facilities include User Interface, Information Management, System Management, and Task Management.

The version 2.0 of CORBA includes an interoperability standard enabling one ORB to pass requests to a different ORB. It comprises bridging mechanisms, translating requests between ORBs, and communication protocols such as the General Inter-ORB Protocol (GIOP) and Internet Inter-ORB Protocol (IIOP).

To promote the importance of CORBA for the healthcare, within the OMG in 1995 the healthcare domain task force CORBAmed was created, which explicitly states as its mission „to improve the quality of care and reduce costs by CORBA technologies". It has already initiated the technology adoption process to standardise interfaces for healthcare domain vertical facilities. The authors are actively involved in these activities.

3. Security features available in CORBA

Security protects information systems from unauthorised attempts to access information or to interfere with their operation. Though the need for incorporating security services into CORBA has been recognised rather early, a comprehensive specification of the proposed security solutions has been available only at the end of 1995. Simplicity, consistency across the distributed co-operating systems, scalability and usability (transparency), flexibility of security policies, independence of security technology, application portability, interoperability, and sufficient performance were defined as goals for an object-oriented security architecture within CORBA. Fulfilling these goals, CORBA provides to users and applications transparently the required security at least on the level of their own environment. In addition, the CORBA security services are also available to security unaware applications.

CORBA provides all important security services, such as identification and authentication, authorisation and access control, security auditing, security of communication including

mutual authentication of clients and targets, integrity protection and confidentiality protection, non-repudiation, and administration of security. Security is defined for domains differing from the point of view of organisational or legal conditions (security policy domains), institutional boundaries (security environment domains), or the technology platforms (security technology domains). Security pertains to various components of the CORBA architecture. A considerable part of security functions is implemented directly through the ORBs or through their bridging mechanisms. Others are confined to transaction services or to additional security services, implemented through specific security-related objects. Finally, security services are also provided by the underlying operation systems and communication services.

Each object service is ultimately requested on behalf of a principal, i.e. an end-user known to the system and separately accountable for the requests it initiates. Unless the principal has been already trustworthy authenticated outside the system (see next section), its authentication is performed by the Principal Authenticator object, associated to each ORB providing a higher level of security. The Principal Authenticator creates for each principal a Credentials object, containing the Principal's privilege attributes, e.g. the access identity, groups to which the principal belongs, roles, security clearance, and capabilities concerning various groups of objects. A security aware target application may obtain attributes of the principal responsible for the incoming request, to make its own authentication-depending access decisions. The information contained in Credentials can be obtained either directly or through the Current, an interface of the Transaction Services, which holds reference to the current execution context at both client and target objects.

The privilege attributes are first needed for making a secure invocation, which is mediated by the ORB. Whether the invocation can take place, as well as the way in which it is mediated, depends on the client and target security policies. Security policies concern such issues as access control, establishing trust in client/target, protection of messages for integrity/confidentiality, time restrictions, or delegation of privileges. If a request initiates a chain of invocations, then the security policies of all objects in the chain are taken into consideration by delegation mechanisms, including all intermediate objects.

As far as access control is concerned, applications can enforce their own access policies. Typically, details of access control are isolated from the application itself, and are implemented through an Access Decision Object, specific to the access policy. In addition, there is an Access Decision Object associated with the ORB and used for the invocation access policy, which is enforced internally by the ORB. The decision whether to allow access to a given function or data depends on the privilege attributes of the initiator of the request, control attributes of the target, and on the execution context. Access policy can be actually shared by a whole domain of objects with similar security requirements. In that case, reference to the corresponding Access Decision Object is available via the Current interface.

Similarly, applications can also enforce their own audit policies, which can be again managed via a domain structure. Each application writes its audit records to an Audit Channel object. One such object is created at ORB initialisation time and is used for all system auditing. Application can use different Audit channel objects.

Finally, CORBA supports optional Non-repudiation services, providing generation and later verification of evidence concerning performed actions and data associated with those actions. The evidence can be generated using either symmetric cryptographic algorithms requiring a trusted third party as the evidence generating authority, or asymmetric cryptographic algorithms assured by public key certificates issued by a certification authority. Keys or other information needed for generating or checking the evidence are available via Credentials.

4. CORBA Security Solutions in the Context of Healthcare Information Systems

Shared care means communication and co-operation between directly or indirectly involved care providers. In this context, in general, a client requests a service from a server. Client and/or server could be a user and/or an application. The guarantee of data security and the reliability and obligation of certain activities are basic conditions for health information systems supporting trustworthiness between physicians and patients, but also between different care providers. Legislation, rules, roles, duties, rights, conditions, and penalties are defined by the security policy. Security threats and risks have to be analysed and assessed. Countermeasures must be evaluated and implemented. These steps must be regularly repeated. Additionally to the availability of information and functions, two kinds of security are needed for the secure invocation of a service or the secure use of an application:

- the communication security, ensuring integrity, reliability (accountability), and confidentiality of communication between authenticated partners, and
- the application security, controlling access rights to the application (functional and data access rights) as well as the reliability of the application functions and data (accountability as non-repudiation).

The access rights depend on the organisational structure within the healthcare institution (mandatory access rights), on the role of the principal within the care process (e.g. caring doctor, therapeutic team, consulting doctor, nurse, administrative clerk), and finally, on the patient's consent. The case of emergency care where the roles of particular principals are not known in advance can be essentially covered using the CORBA identity domain, a special case of security environment domains.

To ensure integrity, reliability, and authentication, strong authentication mechanisms must be used, relying on user-specific knowledge (password, PIN), ownership (electronic identity cards with keys and certificates), or physical properties (such as fingerprint, voice analysis, retina analysis, face analysis). Confidentiality can be provided using symmetric and/or public key cryptographic algorithms. Nowadays, availability, feasibility and cost-benefit relation are promoting chip-cards for security mechanisms in healthcare. Those cards will contain the user's identity, private keys for digital signatures (ensuring integrity and non-repudiation of origin), as well as, if necessary, class keys for group authentication. The latter function could also be provided using the individual authentication, together with directories of group members, their roles and rights. Finally, a trust authority (trusted third party) is needed, to ensure the correctness and validity of keys by certificates, and to provide directory services (public keys for encryption and proof of digital signatures), as well as notary functions.

Functions related to the communication security can be globally organised, whereas the application security related to detailed access rights concerning a particular application can be controlled only locally, by the owner of the data or by the application administrator. In this context, the delegation mechanisms available in CORBA support the above described authentication procedures of security aware healthcare environment. The highly dynamic access rights underlying the access decisions are, in general, enforced by the application via access decision objects and additional services (like time services, account management). Using the various delegation options (simple delegation for end-to-end interactions, composite, combined and/or traced delegation) the middleware can adapt to requirements of different users and establishments. We intend to use this functionality of CORBA for our own environment. Within a project called German Model Trial, aiming at the implementation of health professional cards (HPCs), we are introducing HPCs to improve data security in a regional clinical cancer registry. Our contribution to the project includes also a sophisticated access management.

In distributed co-operating information systems, the underlying middleware provides also some integrative functions. For example, the envisaged CORBA vertical facilities Master

Patient Index, harmonising the patient identification in different applications, and Lexicon Services, supporting and managing terminology and semantics between different systems, provide functionalities supporting the intraorganisational or interorganisational interoperability of different information system components. Since those facilities will support such essential medical functions as electronic health records, archiving systems, clinical or epidemiological registries, they must ensure an adequate level of security.

In current security models, the service providers, including middleware services, are viewed as untrusted, following the basic concept to trust nobody and to organise security mainly by the communicating and co-operating partners. Especially for distributed middleware architectures involving a number of hosts, Varadharajan proposed to install, on each of them, security functions (e.g., encryption/decryption, signatures), a security information base, secure factory objects (objects responsible for creation and deletion of other objects), and secure interfaces [4]. Most of these services can also be provided by functionalities specified in CORBA.

5. Conclusions

The CORBA middleware architecture, as it has been specified so far, provides advanced security services that allow the integration of both security unaware and security aware applications typical for the healthcare area. Special conditions defined in security policies of departments, institutions, organisations, regions, countries, or even the European Union can be specified, to control the middleware security services. The CORBA security solutions are suitable to integrate external security services in healthcare proposed the within TRUSTHEALTH project funded by the Telematics Programme of the EU. Moreover, the integration of such external security services is also possible in coexistence with other middleware approaches, such as DHE and HL7 [2].

6. Acknowledgement

This work was supported within the "Telematics Applications Programme" framework of the European Union, and by the Ministry of Education and Science of the German Federal State Saxony-Anhalt.

References

[1] BLOBEL, B., and HOLENA, M., Advanced Healthcare System Architecture Using Middleware Concepts - A Comparative Study. Deliverable of the HC 1019 Telematics Project HANSA, July 1996.

[2] BLOBEL, B., and Holena, M., Security Aspects of Health Information Systems Based on Middleware Architectures. Submitted to the Int. Journal of Bio-Medical Computing, March 1997.

[3] LEGUIT, F. Interfacing Integration. In Bakker, A.R., et al., Eds., Hospital Information Systems, pp. 141-148. North-Holland, Amsterdam, 1992.

[4] VARADHARAJAN, V., and HARDJONO, T. Security Model for Distributed Object Framework and its Applicability to CORBA. In Katsikas, S.K., and Gritzalis, D., Eds., Information Systems Security, Chapman & Hall, London, 1996, pp. 452-463.

Regarding the included architectural groups, additional references can be obtained from the authors.