# 2

# Improving the security of medical database systems

G. Pangalos[a], D. Gritzalis[c], M. Khair[a], L. Bozios[b] ,

[a] Computers Division, Faculty of Technology, Aristotelian University, Thessaloniki 540 06, Greece
[b] Information Systems Department, AHEPA University Hospital, Thessaloniki 54636, Greece
[c] Department of Mathematics, University of the Aegean, Samos GR-82300, Greece

A database design methodology is presented in this paper which aims to improve the security of medical database systems. The proposed methodology is based on both the discretionary and the mandatory database security policies. In this way the advantages of both approaches are combined to enhance medical database security. The experimental implementation of the proposed methodology in a major Greek hospital is also presented. The implementation has shown that the combined discretionary and mandatory security enforcement effectively limits the unauthorised access to the medical database, without severely restricting the capabilities of the system.

## 1. INTRODUCTION

Security is an important issue when dealing with medical information systems. Current thinking in information systems security is that the issues centre on *confidentiality* (information is only disclosed to those users who are authorised to have access to it), *integrity* (information is modified only by those users who have the right to do so), and *availability* (information and other IT resources can be accessed by authorised users when needed). Medical systems are 'risky' systems with respect to at least these issues [1,2]. Information technology helps to improve significantly the efficiency and quality of health care services. It creates however new situations regarding security that should be dealt with in a thorough and convincing manner. The patient should never doubt for example that information given in confidence to a health professional is collected and stored in medical databases correctly, disclosed only to authorised persons, and used lawfully [1,3]. Introducing security into a medical system is however a balancing process between providing the desirable level of protection of privacy on the one hand and maintaining an adequate level of availability and integrity on the other. The level of security that should be included in a medical information system involves therefore some judgement about the

J. H. P. Eloff et al. (eds.), *Information Security — the Next Decade*
© IFIP International Federation for Information Processing 1995

required level of availability, the dangers associated with the system and the resource implications of various means of avoiding, or minimising those dangers.

Database security plays a significant role in the overall security of medical information systems and networks. This is both because of the role and the nature of the database technology (handles and communicates in most cases stores of valuable and sensitive data) and its widespread use today [1,3]. Several models for database security have been proposed today. However, as it will be seen below, none is sufficient by itself to solve all the problems concerning medical database security. As already suggested by researchers in the field [6,7,8], the combined discretionary and mandatory database security enforcement could effectively limit the unauthorised access without severely restricting the capabilities of the system [1,3]. Such a methodology for enhancing medical database security is described in this paper. The proposed methodology combines the capabilities of the mandatory and the discretionary models in order to increase the level of security in a hospital database system and has been based to a significant extent on the work we undertook in the framework of the SEISMED project of the EU (1). The experimental implementation of the proposed methodology in a major Greek general hospital and the conclusions drawn from it are also discussed.

## 2. MEDICAL DATABASE SYSTEMS SECURITY POLICIES AND THEIR LIMITATIONS

Database security is concerned with the ability of the system to enforce a security policy governing the disclosure, modification, or destruction of information. Given this definition and the general framework of medical database security, we can regard a database as a channel in the sense of communication theory. Then a database security policy states: (i) which type of sub-channels between (groups of) users can be established, (ii) the requirements of the availability of certain facilities of the sub channels, and (iii) the requirements on the (partial) separation and non-interference of sub channels [1]. Seen from this point of view, we can identify two prominent proposals for database security policies, which are most well known and widely used today [1,3]:

a. The *mandatory* (or multilevel) security approach. The need for such a policy arises when a database system contains information with a variety of classifications and has some users which are not cleared for the highest classification of the information contained in the system. The Mandatory Access Control (MAC) is based on the following assumption (constructs): there are users, data items, and a lattice of security levels. The sensitivity label that describes the security level of the data is formed of two parts: the category and the sensitivity level of the information. The category of the information depends on the belonging of the data to a certain party. For example the medical record of a patient can be defined as belonging to a specified ward, so the access by a doctor working at another ward must be prohibited. Only the responsible doctor of a ward is thus permitted to have access to the whole medical record of the patients hospitalised at the rooms of that specified ward. The sensitivity level of the information depends only on the data without any

consideration of the management procedures. For example, the sensitivity levels of the various diseases are defined according to the nature of the illness and the possible social impact of an unauthorised disclosure.

b. The *discretional* (or commercial) security approach. It is designed to enforce a specific access control policy (DAC) and is based on the following assumption (constructs): there are users, (well-formed) transactions, and (constraint) data items. The concept of the 'user role' has a special importance in this policy. A role is a set of actions and responsibilities associated with a particular organisational function. People in an organisation can play various roles which are functions that they carry out in the organisation. Every individual person can play one or more roles. The separation of roles and people playing them provides flexibility in binding people to roles and allows the people that play a role to change without affecting other activities that depend on their roles. The specification of access rights to data can thus be implemented independently of the specific individuals that may be associated within an organisation.

Both approaches have several limitations. An important one in the multilevel security approach is for example that the variety and nature of social roles are not appropriately reflected by the lattice structure of (the usually used) security levels. Linearly ordered "military" authorisations are not expressive enough. The subset lattice of categories cannot deal with degrees of sensitivity, and the widespread Cartesian product of these lattices forces the granting of much higher clearances that are necessary to perform combined tasks. For instance an employee who has to work with classified health data and unclassified salary data must get the clearance (Secret{health data, salary}), and thus he can access salary data. Furthermore it is difficult to express the sensitivity of relationships between data items and to adapt classifications if new tasks evolve [4].

In order to ensure privacy in a hospital environment, a different compartment would be required for each patient when using MAC. In large hospitals this could mean thousands of compartments. Systems supporting several thousand compartments are generally not commercially available. Another limitation of the MAC mechanism is that it lacks the flexibility which is especially required by the medical deontologie. The needs of the medical community require an access control mechanism that is tailorable in terms of the type of access control. Some individuals would have read access to objects (files, tables, columns, rows), others would have read and write access. Addressing this need via MAC and sensitivity labels would require the use of multiple compartments for each patient's medical record (employing one compartment for records which a nurse could access, another for which a physician could access, etc.). Thus instead of one compartment per patient, multiple compartments per patient would be required.

The discretional approach (DAC) also has a number of limitations. Unless restricted by application programs, discretional controls cannot enforce security mechanisms for critical applications, as for example the healthcare applications. This limitation occurs because the burden of enforcing the security policy of the organisation is in the hand of the users that have discretionary access to certain database components (a reason why systems supporting

DAC are often subject to Trojan-Horse attacks [3]). Another problem is that access constraints are not tightly bound with the data. Thus a user who is allowed only read access to a data object in DAC would still be able to make a copy of that object and pass it on to some other user. In the above medical scenario, DAC would for example be unable to prevent a subject, acting on behalf of a consulting physician, from making copies of a patient's record which could subsequently be passed to another individuals.

None of the two approaches alone is therefore sufficient by itself to solve all the problems concerning medical database security.

## 3. DEFINITION OF THE PROPOSED DATABASE SECURITY POLICY

The proposed database security policy aims to enhance the overall security of medical database systems by combining the advantages of both approaches. An approach that has been a part of the model of Bell and LaPadula, and is also required for higher security classes (A- and B-level) according to the TCSEC (1). Users are classified for this purpose into a number of categories according to their need-to-know requirements. A User Role Definition Hierarchy is used to characterise the different types of users who access the medical database. User identification and authentication is first appropriately handled. Each group of users is then capable of accessing the database in a predefined way that minimises the risks to the system and at the same time assures the satisfaction of the performance requirements and the needs of the users.

A user who wants to access the database has to pass through a number of layers (figure 1). All users asking for a certain authorisation to access the database will first pass through the first layer. Based on its type (section 3.2), that user will then either pass through both remaining layers (both discretionary and multilevel controls), or only the third one (multilevel controls only).

1. First layer: In this layer the identification and authentication of a particular user is performed by the security mechanisms of the host system. The host system is defined as the operating system, or as the combination of the operating system and the network operating system. Security mechanisms can be exploited, or enhanced at this level by retrofitting security features, that is, by adding new security functions through the introduction of add-on packages, or security special-purpose hardware.
2. Second layer: Users are required in this layer to access the database via a user-role-based discretional security model. Depending on the actual role the user is going to exercise in the application, he will have the authorisation to see a view, which is predefined by the administrative users and the security administrator in the way described below, and exercise just the authorised access rights. The definition of this step is done using the (DAC) security policy.
3. Third layer: Users are required in this layer to access the database via a multilevel security model. Depending on the category of the user, or in other words its responsibility framework, the specific user will get just the part of information that he needs to perform his task. This is enforced by using the (MAC) security policy,

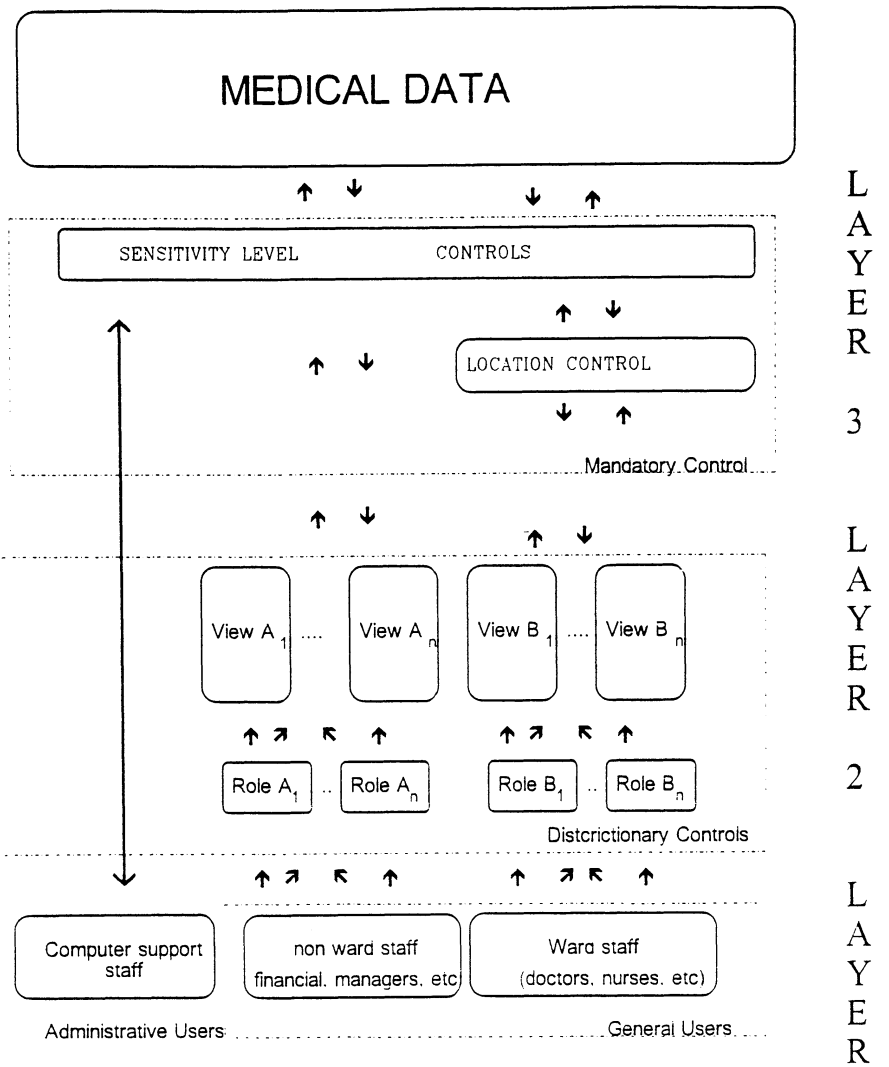


Figure 1. An overview of the proposed methodology

depending on the categories of the information that are part of the security label of the data. Subsequently, depending on his level of clearance, he will have access to a certain part of the database which is defined according to the dominance relation (i.e. the sensitivity level of the information which can be accessed by the user is less, or equal to the his clearance level).

## 4. CLASSIFICATION OF THE USERS

Users are classified into two major categories, according to the nature and security requirements of the tasks they perform: the administrative users, and the general (non-administrative users). The administrative users, in which we include the developers and the maintenance support staff of the applications (DBAs, programmers, etc.), do not have to pass through any discretionary security controls (second layer) but they have access directly to the database, depending on their clearance level (multilevel security controls). The general users have to pass through both discretionary security controls (second layer) and multilevel security controls (third layer). General uses are divided again in two categories: the ward staff (doctors, nurses, etc.) and the administration support staff (financial, managers, etc.). Because of the nature of their work, the medical ward staff should have access only to the information related to the patients that are hospitalised in the wards, or in certain rooms that they are responsible for. This is assured during the logging in procedure by the introduction of the location control in the sensitivity label of the data. On the other hand, the administration support users, due to the nature of their work which is not usually restricted to a specific part of patients, are not necessarily restricted by the location control layer, since they can be granted by the administrative users access to certain, or all locations.

The major advantage of using this method for representing of the users, is that the granting and revoking rights are only given to the administrative users. Using the multilevel security policy for the representation of these users, the problem of revocation of authorities (cascading, or normal) that is found in the discretionary security policy is eliminated. Another advantage of this methodology is that the definition of the views is spread among administrative users. A more detailed description of our methodology can be found in [3].

## 5. THE USER ROLE DEFINITION HIERARCHY

The User-Role Definition Hierarchy (URDH) is used to characterise the different kinds of individuals (and groups) who all require different levels of access to the application (figure 2). The responsibilities of the users are characterised into three distinct levels of abstraction for the URDH: user roles, user types and user classes. User roles allow the security designer to assign particular privileges to individual roles. User types characterise common responsibilities among related user roles. Finally the different user types can be grouped into one, or more user classes. A four step methodology which has been described in detail in [6] has been used for defining the user roles (types and classes), the methods assigned to them and the conditions under which they work.

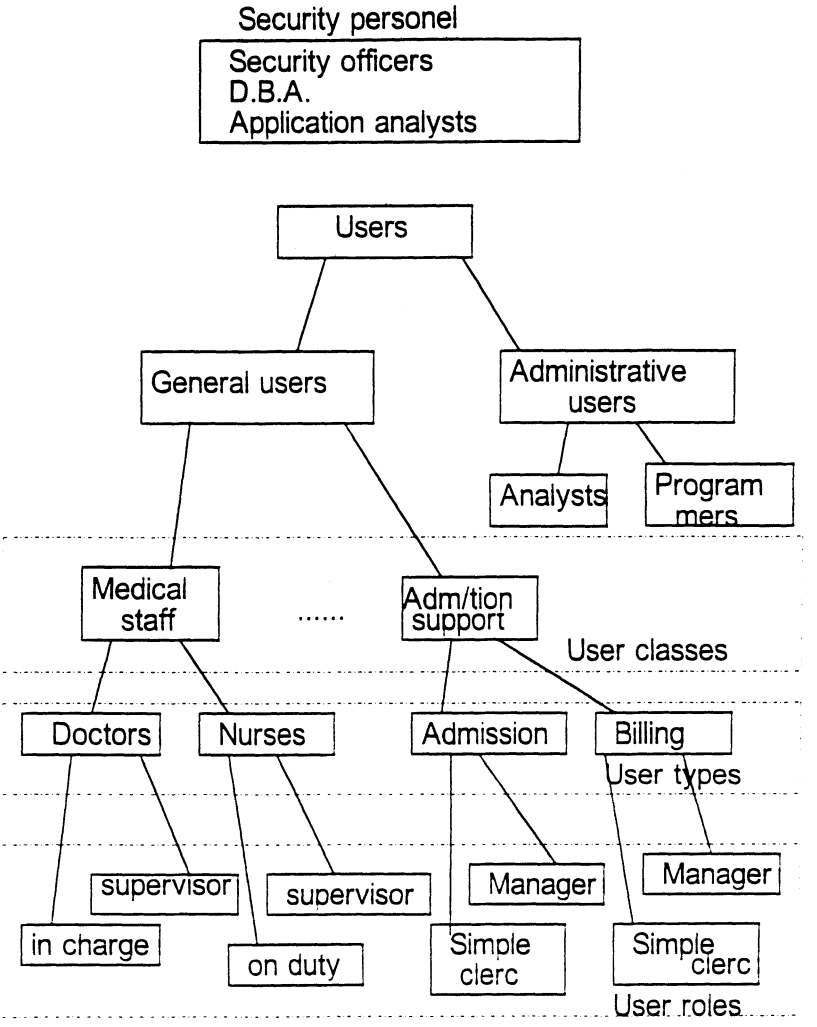


Figure 2. The User Definition Hierarchy

## 6. THE EXPERIMENTAL IMPLEMENTATION

The AHEPA University Hospital, which has been used as our test-bed, is a general hospital which is part of the Aristotelian University of Thessaloniki. The following figures describe briefly the hospital: 16 clinics including the reference and hospitalisation centre for AIDS patients from all Northern Greece; 40 laboratories; a radiological department including M.R.I., C.T., U.S., D.S.A, X-rays, etc.; a nuclear medicine department (SPECT, Gamma-Camera); 705 beds; 520 medical doctors including consultants; 762 nursing personnel; 466 personnel for financial and general support; 28,000 inpatients per year; 2,500 surgical procedures per year; 107,000 outpatients per year; 2,345,000 laboratory tests per year (1993).

### 6.1 Definition of the primary data sets

A patient record is usually characterised by high complexity and heterogenity in both the nature and the sensitivity levels of the different data sets included in it. Organising these data in a structured manner is necessary for the development of the appropriate user views, which in its turn is a required step in the design and implementation of the hospital information system. The exact set of data groups obviously depends on the particular hospital system under study. Based on the experience obtained through the pilot implementation [3,4] and the related research work, we can however identify the following typical set of basic data groups, which will be used to demonstrate the process:

*Administrative information:* It includes information related to admission data, in-charge and out-charge information, patient follow-up guidelines, etc.

*Non-medical historical information:* It includes some of the patient 'life-style' related information, like smoking habits, working environment details, etc.

*Social information:* It includes information related to 'social data' concerning the patient, like religion, type of work, sexual life style, etc.

*Personal demographic information:* It includes patient data like name, ID number, address and other types of information that may lead to the direct identification of the patient.

*Non-personal demographic information:* It includes demographic information about the patient that may be used for statistical purposes. This information could not lead to the identification of the patient.

*Insurance information:* It includes all insurance related information.

*Diagnosis:* It includes information related to symptoms and patient illness, as defined by the responsible physicians.

*Examination request:* It refers to the doctor orders for specific laboratory tests, other examinations, etc.

*Examination result:* It refers to the laboratory and radiological test results.

*Treatment Data:* It refers to the data about the treament given to the patient. It may be divided to the following categories: *pharmaceutical, clinical, operational, radio therapy, physical therapy, diet and mental treatment.*

*Use of special materials:* It refers to some specific data like the ones related to heart operations, special materials, etc., that can lead to a direct conclusion of the patient illness.

*Billing information:* It relates to all the billing oriented (financial) information of a patient.

**6.2 Definition of the user roles**

Physicians and the other staff in a hospital department can have different responsibilities. A representative [1,5] hierarchy, which has been used for the identification of the need-to-know requirements of the various types of users in the experimental implementation (which is also representative for most Greek hospitals), is the following:

*Head doctor:* He is responsible for the entire department. He is not personally involved in the everyday medical practice but he administers the medical practice of all the therapists in his department.

*Responsible doctors or therapists:* They make diagnosis, admissions, and specify treatments during the normal working hours of the day.

*On duty doctors:* He is the doctor responsible outside the normal working hours of the day (e.g. during the night). During his duty time he has all the responsibilities that are normally given to the therapists.

*Head nurse:* She has administration activities. These relate to the examination requests, pharmaceutical prescriptions, clinical treatments, operations workflow (e.g. waiting list for the surgical operations), etc.. She orders the treatments and special materials according to the therapist request and to the hospital rules.

*Nurses:* They are responsible for providing the daily care to the patients. In our experimental implementation we have assumed that a nurse does not need to know any sensitive personal patient information (since by using the hospital id-number and the bed number of the patient she is able to perform her tasks).

*The paramedical staff:* They collect specimens and perform different tasks which include body and blood tests, radiology tests and all the paramedical treatments ordered for a specified patient.

*The paramedical doctors:* They are responsible for the laboratory tests, (especially those that require human intervention) and the evaluation of the results. In most of the cases (especially in complex radiological examinations like CAT or MRI) these doctors cooperate with the therapists for the definition of the final diagnosis and the treatment plan.

*Registration staff:* They are responsible for the collection of the administrative, social, personal and non personal, demographic and insurance information about the patient. The precise information that is needed by the financial office about the insurance coverage of the patient is included within their responsibilities.

*Financial staff:* They are responsible for the update of the financial data, e.g. the calculation of the hospitalisation costs. They perform their tasks by using the internal hospitalisation number and the insurance information without having access to the identity and other personal data of the patient.

**6.3 Definition of the permitted actions**

The basic well defined actions in most DBMSs are 'Select', 'Insert', 'Update' and 'Delete'. In a medical database system environment however, deletion is usually prohibited in the sensitive components, for auditing and follow-up reasons [1]. In our experimental implementation we use a logical deletion procedure, instead of the physical updating and

deleting. This procedure activates a flag that can have three states: 'Inserted', 'Cancelled' and 'Executed'. 'Inserted' means that the action is under execution (is valid), 'Cancelled' means that the action is cancelled, or stopped and 'Executed' means that the action is performed. For example during the insertion of a laboratory test request this flag is turned to 'Inserted'. If the act is committed it will be turned to 'Executed', and in the case that the doctor wishes to stop this test the flag will be turned to 'Cancelled'. This procedure is followed in the diagnosis field and all the other action fields (the fields that imply actions). The identity of the initiator of the action (Insert, Cancel, and Execute) is logged, for a possible investigation of the medical workflow (audit).

### 6.4 The experimental implementation

Because of technical difficulties it has not been possible yet to implement our system on a trusted version of the DBMS (which would support directly the use of both mandatory (MAC) and discretional (DAC) database security policies). For this reason and in order to simulate (for research purposes) the way our model works, we implemented temporarily the sensitivity label concept indirectly, as a special field in the record, using the ORACLE DBMS version 7, along with the existing data schema and system architecture. The system is implemented on a distributed database environment, UNIX-based servers and workstations, TCP/IP communication protocols and client/server applications. We have used the provided grouping features of the DBMS (Oracle 7) to define the roles and the related individual users. In the runtime environment, the interface between a user and the application depends on: (i) the physical access to the terminal, (ii) the node, and (iii) the identification and the authorisation procedures supported by the O.S.. During execution the sensitivity level is passed as a dynamic parameter to transparently reduce the user view based on his task in the application, or his need-to-know requirements principle.

The proposed security policy described in section 3.2 above has been implemented. A security schema was developed for this purpose which has been based on the multilevel access control (MAC) policy model (table 1). This has been used for the representation of the user roles, the sensitivity clearances of each role and the corresponding data sets. This classification has been based on the decision of the security officer, according to the hospital and departmental security regulations. A special, additional column has been used for the internal representation of the sensitivity label of the data (since our DBMS does not support directly MAC).

A number of interface applications (user views) have also been constructed, based on the discretional access control (DAC) policy. A number of access tables have been developed for this purpose, one for each type of user. An example of such a table is given in table 2. A summary of the complete security schema (access policy) implemented is given in table 3. This schema is used for the representation of the views of the different user roles, and the modes in which these users are permitted to access the specific data groups. The schema is based on a strict appliance of the need-to-know principle.

| User roles | Sensitivity / clearance - levels | Data sets |
|---|---|---|
| Personal doctor<br>Registration staff<br>On duty doctor<br>Head doctor | 5 | Personal demographic |
| Statistical personell<br>Head nurse<br>Paramedical doctor | 4 | Non personal demographic<br>Social info.<br>Non-medical historical |
| Paramedical staff<br>Billing staff | 3 | Insurance<br>Radio-therapy treatment<br>Physical-therapy treatment<br>Diet<br>Mental treatment<br>Special materials<br>Exam requests and results |
| Nurses | 2 | Diagnosis<br>Pharmaceutical treatment<br>Clinical treatment<br>Operational treatment |
| Others | 1 | Administrative info. |

Table 1: The security schema developed according to the MAC access policy.

| Access-modes<br>Data-sets | select | insert | cancel | execute |
|---|---|---|---|---|
| Administrative info. | x | | | |
| Non-medical historic | x | | | |
| Social info. | x | | | |
| Personal demographic | x | | | |
| Non personal demographic | x | | | |
| Insurance info. | | | | |
| Diagnosis | x | x | x | |
| Examination requests | x | x | x | |
| Examination results | x | | | |
| Pharmaceutical treatment | x | x | x | |
| Clinical treatment | x | x | x | x |
| Operational treatment | x | x | x | x |
| Radio-therapy treatment | x | x | x | |
| Physical therapy treatment | x | x | x | |
| Diet | x | x | x | |
| Mental treatment | x | x | x | |
| Use of special materials | x | x | x | |

Table 2. The user view for the ' responsible doctor-therapist' type of user (DAC).

| **User-roles, Data sets, access-modes** | Head doctor | Therapist doctor | On-duty doctor | Head nurse | Nurse | Para-medical staff | Para-medical doctor | Registration staff | Billing staff |
|---|---|---|---|---|---|---|---|---|---|
| Administrative information (s,i,u) | s | s | s | s | s | s | s | s,i,u | s |
| Non-medical historic (s,i,u) | s | s | s | s | | | s | | |
| Social info. (s,i,u) | s | s | s | | | | | s,i,u | |
| Personal demographic (s,i,u) | s | s | s | | | | | s,i,u | |
| Non personal demographic (s,i,u) | s | s | s | | | | | s,i,u | |
| Insurance information (s,i,u) | | | | | | | | s,i,u | s |
| Diagnosis (s,i,c) | s | s,i,c | s,i,c | s | s | s | s | | s |
| Examination requests (s,i,c,e) | s | s,i,c | s,i,c | s | | s,e | s,i,c,e | | s |
| Examination results (s,i,c) | s | s | s | s | | s | s,i,c | | s |
| Pharmaceutical treatment (s,i,c,e) | s | s,i,c | s,i,c | s,e | s,e | s | s | | s |
| Clinical treatment (s,i,c,e) | s | s,i,c,e | s,i,c,e | s,e | s,e | | s | | s |
| Operational treatment (s,i,c,e) | s | s,i,c,e | s,i,c,e | s | s | | s | | s |
| Radio-therapy treatment (s,i,c,e) | s | s,i,c | s,i,c | s | | | | | s |
| Physical therapy (s,i,c,e) | s | s,i,c | s,i,c | s | | | | | s |
| Diet (s,i,c,e) | s | s,i,c | s,i,c | s,e | s,e | s | s | | s |
| Mental treatment (s,i,c,e) | s | s,i,c | s,i,c | s | | | | | s |
| Use of special materials (s,i,c,e) | s | s,i,c | s,i,c | s,e | | | | | s |

Table 3: The security schema developed according to the DAC access policy.

## 7. DISCUSSION

Based on the experience from designing, implementing and using the above system in the testbed hospital the following remarks can be made:

1- No observable performance overheads have been noticed (fig. 3). This was to a significant extend the result of the correct indexing and database table clustering. Also the complexity of the applications has not affected the overall system performance (fig. 4), mainly because of the use of the Client/Server model.

2- Some additional physical design storage and programming overhead was required (as was expected), due to the extra columns used for sensitivity levels and the implementation of application and database triggers. These triggers are used to preserve the security countermeasures during the access of the data.

3- A source of several problems has been the fact that the Greek security legislation and the internal hospital regulations lack an exact definition of the users' duties and responsibilities in the overall system, in a feasible way to make more efficient the design of the security hierarchy.

4- Because of the limited flexibility of the information system (hard structured) some problems were encountered caused by the overlapping duties of the medical personnel and the unprogrammed changes of shifts between the working personel.

5- There have been some problems on the information system acceptance by the end users. These were however mostly due to their ignorance of the security threats. This can be solved by organising special security and privacy oriented seminars.

## 8. CONCLUSIONS

An integrated secure design methodology for the enhancement of database security and its experimental implementation in a hospital environment has been presented in this paper. The proposed design methodology is based on both the discretionary and the mandatory database security policies. In this way the advantages of both approaches are combined in order to enhance medical database security. The experimental implementation of the methodology in a major Greek hospital has also been presented. The implementation has shown that the combined discretionary and mandatory security enforcement effectively limits the unauthorised access to the medical database, without severely restricting the capabilities of the system.

## 9. REFERENCES

1 G. Pangalos, Security in medical database systems, EEC, SEISMED project report, No. INT/S.3/92, (1992).

2 J.V. Marel, A.B. Bakker, User accessrights in an intergrated hospital information system, IFIP-IMIA, North-Holland publ., (1988).

3 G. Pangalos, A. Pomportis, M. Khair, L. Bozios, Development of secure medical database systems, proc. DEXA'94, (1994). Also in Lecture notes in computer science, Springer-Verlag, (1994).

4 G. Pangalos, M. Khair, L. Bozios, Emhancing medical database security, to appear in Journal of Medical systems, (1994).

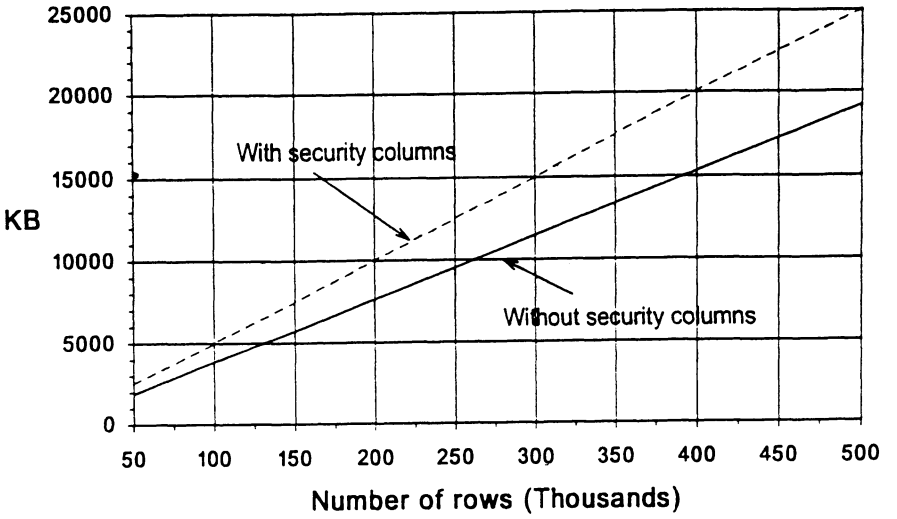


Figure 3. Example of table storage overhead due to additional security structure (columns)

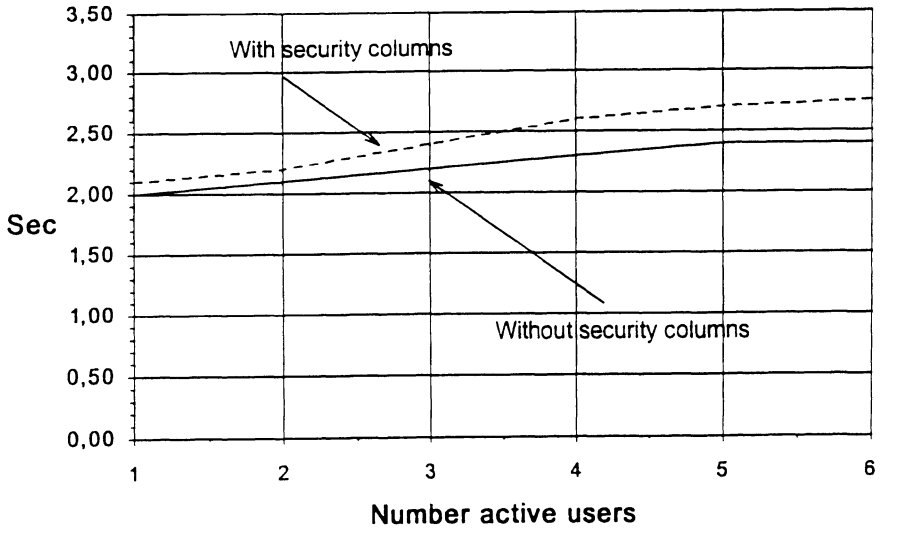


Figure 4. **Response time overheads due to additional processing**

5 S. Furnell, G. Pangalos, P. Sanders, M. Warren, A generic methodology for health care data security, to appear, Medical Informatics, (1994).
6 T. Ting , A user-role based data security approach, Database security, Landwehr (ed), (1988).
7 D. Denning, An Evolution of views,Research Directions in Database Security, Lunt (ed), (1992).
8 L. Notargiacomo, R. Graubart, Health Delivery: The Problem Solved?, Database Security 1V: Statua And Prospects, Jajodia and Landwehr(ed), North Holland publ., (1991).