MEDICALLY ADAPTIVE ROLE BASED ACCESS CONTROL MODEL (MAR-BAC)

Naim Alperen Pulur

Submitted to the Graduate School of Engineering and Natural Sciences
in partial fulfillment of
the requirements for the degree of
Master of Science

Sabancı University

August, 2015

MEDICALLY ADAPTIVE ROLE BASED ACCESS CONTROL MODEL (MAR-BAC)

APPROVED BY:

Prof. Dr. Albert Levi	.....
(Thesis Supervisor)	
Asst. Prof. Dr. Mordechai Shalom	.....
Asst. Prof. Dr. Kamer Kaya	.....

DATE OF APPROVAL: .....

MEDICALLY ADAPTIVE ROLE BASED ACCESS CONTROL MODEL (MAR-BAC)

Naim Alperen Pulur

Computer Science and Engineering, Master's Thesis, 2015

Thesis Supervisor: Prof. Dr. Albert Levi

Abstract

The development of technology gives opportunity to reach information in a reasonably short amount of time. Ease of access to information does not only create positive consequences, but also provides an easy way to access to information by unauthorized parties. As a result, the requirement of protecting data from different aspects of security turns into a significant issue of the information systems. Another issue in such systems is safeguarding the access permissions in order not to allow public accesses to private data. Protecting the data from disclosure, tempering or destruction as well as prevention of unauthorized use of any resource are important aspects of the security in medical environments since the medical data is private data.

In this thesis, we introduce a novel access control mechanism in order to safeguard privacy of medical data of patients in dynamic environments. Our access control model, called MAR-BAC (Medically Adaptive Role Based Access Control), takes advantages from role-based access control (RBAC) and criticality-aware access control (CAAC). Our original approach allows the medical professionals with different roles to be granted access to medical records of patients automatically and without explicit request in case of a medical emergency. In this context, we design secure and privacy aware protocols from initial login to patients' medical data transmission and retrieval by the medical professionals. We mostly take a formal approach in our access control model definitions

and procedures. The medical awareness feature of our MAR-BAC model comes from the fact that medical data of the patients are analysed in near real-time. Each such analysis yields automatic updates in the access control rules for the sake of urgent medical attention. We carry out simulation based performance evaluation to determine the delay characteristics of our MAR-BAC model. We also analyse the scalability of the system. Our results show that MAR-BAC scales linearly under moderate system load. Again under moderate load and in a hospital with 500 inpatients, the maximum end-to-end delay to react a medical emergency is less than 12 seconds.

Tıbbi Şartlara Uyum Sağlayabilen Rol Tabanlı Erişim Denetimi

Naim Alperen Pulur

Bilgisayar Bilimleri ve Mühendisliği, Yüksek Lisans, 2015

Tez Danışmanı: Prof. Dr. Albert Levi

Özet

Teknolojinin gelişimi, bizlere bilgiye oldukça kısa bir sürede ulaşma şansı vermektedir. Bilgiye ulaşmanın kolaylığı sadece pozitif sonuçlar yaratmamakta, aynı zamanda yetkisi olmayan kişilerin bilgiyi ele geçirmesini kolaylaştırmaktadır. Bunun bir sonucu olarak, veriyi farklı güvenlik açılarından korumanın gerekliliği, bilgi sistemlerinin önemli bir sorunu haline gelmiştir. Bu sistemlerdeki bir başka husus ise özel bilgilerin herkes tarafından erişilmesini engellemek adına erişim izinlerini korumaktır. Tıbbi veri de özel bilgi kapsamında olduğundan ötürü, verinin yetkisiz kullanımını engellemenin yanısıra, veriyi açığa çıkmaktan, değiştirilmekten ve tahribattan korumak da tıbbi ortamlardaki bilgi güvenliğinin önemli gereksinimlerindendir.

Bu tezde, değişken ortamlardaki hastaların tıbbi verilerini korumak amacıyla yeni bir erişim denetimi mekanizması önerilmiştir. Erişim denetimi modelimiz MAR-BAC (Tıbbi Şartlara Uyum Sağlayabilen Rol Tabanlı Erişim Denetimi), rol tabanlı erişim denetimi (RBAC) ve kritik durumun farkında olan erişim denetimi (CAAC) modellerinin avantajlarını kullanmaktadır. Özgün yaklaşımımız, acil vakalarda, değişik rollerdeki tıbbi uzmanların tıbbi hasta kayıtlarına açık bir istek olmaksızın otomatik olarak erişim kazanmasına imkan sağlamaktadır. Bu kapsamda, başlangıçta oturum açmaktan, hastaların tıbbi verilerinin iletimine ve tıbbi uzmanlar tarafından erişimlerine kadar güvenli ve gizlilik bilinçli protokoller tasarladık. Erişim denetimi model tanımlarımızda ve yöntemlerimizde çoğunlukla biçemsel bir yöntem izledik. MAR-BAC modelimizin tıbbi farkındalık özelliği, hastaların tıbbi verilerinin yaklaşık gerçek zamanlı

olarak analiz edildiğinden gelmektedir. Bu analizlerin her biri, acil tıbbi müdahale adına, erişim denetim kurallarının otomatik olarak güncellenmesiyle sonuçlanmaktadır. MAR-BAC modelimizin gecikme karakteristiklerini belirlemek için simülasyon tabanlı performans değerlendirmesi uygulanmıştır. Aynı zamanda sistemin ölçeklenebilirliği de analiz edilmiştir. Sonuçlarımız MAR-BAC sisteminin ortalama sistem yükü altında lineer bir şekilde ölçeklendiğini göstermektedir. 500 adet yatan hastaya sahip bir hastanede ve ortalama yük altında, tıbbi bir aciliyete, uçtan uca tepki verme süresi 12 saniyeden daha azdır.

to my beloved family...

Acknowledgements

This thesis would not have been possible without the support of my supervisor, committee, friends and family.

Foremost, I would like to express the deepest gratitude to my thesis supervisor Prof. Dr. Albert Levi. The presented work existed and developed with the help of his knowledge as well as his guidance, encouragement and patience. I also would like to thank my thesis jury, Asst. Prof. Dr. Mordechai Shalom and Asst. Prof. Dr. Kamer Kaya for their valuable suggestions and inquiries.

I am thankful to all the members of our Cryptography and Information Security Lab for the great environment they provided in terms of both research and friendship. I'm also grateful to my project partners Duygu Karaoğlan Altop and Dilara Akdoğan. They supported me whenever I need help. Every one of them is important to me, but Ecem Ünal has a special place among them. I am beyond grateful to her presence when I needed motivation the most; her unconditional moral and material support aided me during my studies. In addition, I would like to thank my old friends Gamze Tillem, Dilara Akdoğan and Berkay Dinçer for their presence. They give me the opportunity to throw my lot with them more than 7 years. I also would like to my friend Dr. Elif Güven for her assistance in understanding medical environments.

This thesis has been supported by TÜBİTAK under grant 114E557.

Last, but not least, I would like express my special appreciation and thanks to parents; my aunt Suzan Özel and her husband Ahmet Özel, my father Mehmer Ali Pulur and his wife Şirin Pulur, and of course my sister İrem Pulur and my brother Can Pulur. I am thankful especially my cousins Çağrı Özel and Asst. Prof. of Accounting Naim Buğra Özel and my grandmother Kadriye Pulur. I would not be here without the unlimited love and support of my parents provided throughout my life.

Contents

List of Algorithms

List of Figures

List of Tables

1 Introduction

Access control has been an important security service since certain resources are not open for public usage. In a cyber environment, those resources should be reachable by a limited number of subjects and those subjects must be explicitly defined within the system. In information management, subjects should only be able to access to allowed resources; the others' resources should not be accessed.

With increases in the growth of wireless networks, mobile devices and other technologies involved in remote access to resources, management of the access becomes more important. This is due to rapid increase of the number of objects and the number of subjects defined within the system. Therefore, access control systems should not only perform correctly but should also work efficiently in order to operate with an adequate response time. Moreover, if the information is considered to be sensitive, then it requires to be managed with a secure model that should not leak any information to the foreign parties.

Role based access control (RBAC) is an important model of access control paradigm. The RBAC model introduces a mapping between roles and permissions instead of identities and permissions. The main advantage of this model is less administrative overhead as compared to the traditional access control models.

In a medical environment, utilising RBAC could be useful for retrieving information and granting access rights. However, pure RBAC could not assist medical experts in emergency conditions. Consider a scenario that a medical sensor is retrieving information from a patient's body and sends this data to hospital server. If this information is retrieved whenever the subjects' request, then the system might miss some emergency conditions that happen at unrequested times. Such RBAC based systems take

the security as the main concern, but become unaware of medical conditions and emergencies. However, a medically aware system should not only control the accesses of the information, it also needs to be aware of medical condition of patients, especially in emergencies. Under emergency conditions, the system should be able to respond at real time according to the situation in order not to affect negatively patient's health condition. To address these issues, the system should also analyse and interpret the medical information and adapt access rights accordingly.

1.1 Our contribution in a nutshell

In this thesis, we propose MAR-BAC, Medically Adaptive Role Based Access Control mechanism for healthcare management. In our model, patient's medical data is going to be interpreted and analysed in real-time. The purpose of this analysis is to be aware of patients' medical condition. Under emergency conditions, the system should trigger an alarm in order to take responsive actions with the assistance of the analysis. As a result of this analysis, if a critical condition has to be responded by a medical expert, access control policies will dynamically change the access rights on the patient's medical data. Medical experts, who are specialized with the particular disease or complications, are going to be selected and notified about patient's emergency condition. Hence, dynamic changes about access rights of patient's medical data is performed according to the emergency conditions. Once the emergency condition passes over, access rights are restored to the original state.

Our MAR-BAC model is able to transmit the sensed medical record of a patient. The transmission of the data is secured with the establishment of a secure channel. Moreover, our model gives the opportunity to access the medical information under the regulation of access control policies. Under emergency conditions, it provides dynamical changes in access rights of medical experts in order to recover patient's health condition. It does not only dynamically change access rights of the medical experts, but also notifies those personnel for the sake of fast response to the condition.

We performed simulation-based performance analysis of our MAR-BAC system using different metrics and parameters. Performance results show that our system causes

reasonable end-to-end delay, although it varies with the number of subjects. Moreover, the delay introduced by security and privacy related processing is much less than the other delay components.

1.2 Outline of thesis

The rest of the thesis is organised as follows. Section 2 will give the background information. Related work and problem statement can be found in Section 3. The proposed access control model definition and its protocols are explained in Section 4. Performance evaluation is detailed in Section 5. Finally, the thesis is concluded in Section 6.

2 Background Work

Firstly, this section briefly discusses access control models in historical perspective. Subsequently, privacy and diagnosis of the medical data is also mentioned in this section. After the importance of privacy is stated, the security mechanisms which are used in this work are shortly explained.

2.1 Access Control

Today's information management systems should protect resources against unauthorized disclosure (secrecy) and unauthorized or improper modifications (integrity), while at the same time ensuring their availability to legitimate users (resistant to denial-of-service) [3]. This is a significant requirement because any leakage of information about an organization's consumers, strategic plans or products to a competitor may result in financial, reputation losses and legal liability [4]. Therefore, access rights defined on resourced should be controlled in order to authorize acces only to legitimate users. This process is termed *access control*. Decision taken for an access request is generally needed to be predefined. This predefined decision rules implements regulations are so called *security policy* of the access control. *Permission* (or privilege) is authorization to perform an action on the system [5]. Subjects are able to access objects according to the permissions defined within the access control system.

Two important definitions related to this concept are *objects* and *subjects*. An object is the smallest accessible resource on a computer system [5]. Objects can be any data or services which are accessible to predefined subjects. The subjects, which are able to access objects, are selected according to the regulations defined in the security policy. The term *user* is used for the people who are eligible to access certain resources on

a access control model. However, user and subject does not mean same entity in this concept. More precisely, users are subset of subjects. In other words, a user is a subject but not visa versa. Subjects can also be processes in a computer. A user could have multiple subjects in operation. Consider the example a user in an operating system would like to read a certain file. While reading the file, (s)he may also would like to modify another file as well. Therefore, each of user's read and write requests are referred to a different processes, namely distinct subjects.

In the following subsections, some of the important access control models are explained.

2.1.1 Discretionary Access Control (DAC)

Discretionary access control (DAC) [6–9] refers to access that allows users to alter the features of the object as well as to specify whether the object is accessible to other users. Access control is maintained via the following way. One or more users can control the decision of the access to certain object. Those users are generally the owners of the object or decision making is delegated from creator of the object. The controllers decide about access rights on the object so that which subjects are able or not allowed to access the resource. This mechanism is called DAC model and it is also called an identity based access control (IBAC) [10]. As a result, control over accesses depends on the identity of the requester and access control policy states what the requester is allowed to do.

General implementation of DAC model is based on the users, who generate the resources or creators of the objects, establish the rules over the objects. In other words, the users, who own the resources, are able to grant privileges to other users defined within the system. Users can also revoke the permissions from accesses originated from other subjects [3]. Therefore, privileges can be utilized in a two-way manner. It can be granting access to or rejecting access from other subjects.

Access matrix model is the early step of the DAC. It is first proposed by Lampson [11] for operating systems file system management. Access matrix model states are defined with subject, object and access matrix. Matrix rows are defined as subjects

and columns are referred to the objects. In Figure 1, it can be seen a single entry in the matrix corresponded to permissions given to subject (which is defined in row) over resource (which is defined in column).

Subject \ Object	File A	File B	File C	Program 1
Alice	Own, Read, Write	Read, Write		Execute
Bob	Read		Read, Write	
Carol		Read		Execute, Read

Figure 1: Access Matrix Model

However, this matrix is going to have too many entries if access control should be maintained for a large number of subjects. Generally, the matrix end up as sparse. Sparse means that, most of its cells are empty. Therefore, it will consume lots of space. Figure 2 gives three different practical model in order to solve the problems of access matrix model.

Authorization Table Each entry in the table consists of subject, action and object. It defines which subject is able to perform which action over an object.

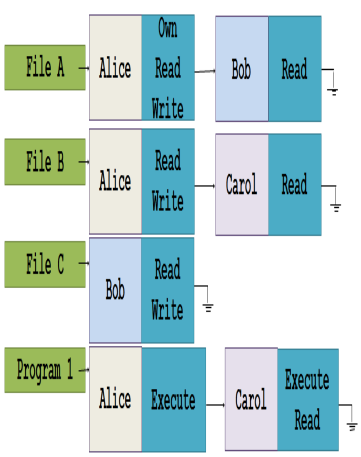
Access Control Lists Each object has a list of subjects who are able perform an action over that object. List nodes contains both subject information and also which actions are able to performed by the subject.

Capability Each subject has a list of objects. In each list element, object and actions able to be performed by the subject is defined.

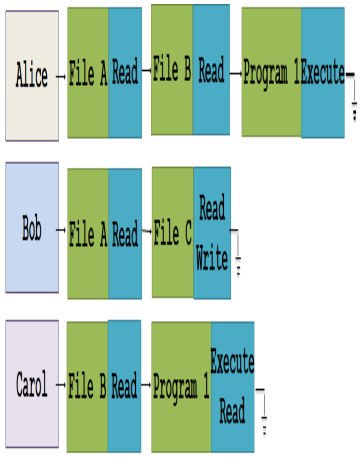
In the authorization table model, it is hard to find whether given subject, action, object tuple exists within the table or not. It is the same as finding an element defined in a linked-list [12]. Access control lists take advantage from finding access regulations defined over objects. On the other hand, in those lists, it is not easy to find access policies defined over subjects. One needs to iterate over each rule defined in access control list in order to find all rules defined over subjects. Capabilities distinguishes the problem of finding all access rules defined on a single subject because it basically maintains lists which are mapped over subjects. However, this time finding all access rules over a single object requires to iterate over each rule within the capabilities. In a capability based system it is mentioned that system is vulnerable to forgery (unau-

Subject	Access Mode	Object
Alice	Own	File A
Alice	read	File A
Alice	write	File A
Alice	read	File B
Alice	write	File B
Alice	execute	Program 1
Bob	read	File A
Bob	read	File C
Bob	write	File C
Carol	read	File B
Carol	execute	Program 1
Carol	read	Program 1

(a) Authorization Table



(b) Access Control Lists



(c) Capabilities

Figure 2: Improvements over Access Matrix Model

thorized usage of access rights) [3]. If user acquire its capabilities over a system, it can crate a copy of the capabilities and maliciously give those to a third party. Since the third party has the capabilities for given system, it can request access as defined in the capabilities from the system. Another problem for capabilities is revocation of capabilities which are already released from the system. If a user is gets its capability list from the system, revocation does not directly modify the capabilities taken by the user.

2.1.2 Mandatory Access Control (MAC)

Mandatory Access Control or MAC security model is one of the oldest access control mechanisms. Main objective of this model is to protect system resources against inappropriate or undesired user access [13]. It restricts access to objects which are requested by subjects. The entities or subjects, which require to access certain objects such as data files, devices, systems, etc., must be given access rights explicitly [14]. As a result of this requirement, the access is centrally controlled by security policy administrator or system administrator. System administrator specifies which entities in the subjects set are able to reach resources on an individual basis. The model was formalized with the requirement of restricting individual resource owners have to be granted or denied

access to resource objects in file systems. The records which subjects are able to access specific objects are stored in access matrix [15]. The security policy defines which type of accesses are going to be granted for each entity [16].

The subjects are restricted with the security policy which is controlled by system administrator. This means even a subject, which is the owner of a specific object, has limited access over the object. An analogy for this can be a multilevel system for military or governmental documents and files. Some of the files must be restricted with limited access. The restriction may be the number of entities which request to access the data, meaning that certain subjects are able to see contents of data. Another limitation is, even subjects have right to access data, they may not be able to see data as a whole so that they can only read file or can read some part of the file.

There is a branch of mandatory access control called the Bell-LaPadula. This model basically focuses on the confidentiality of the objects. It utilise access classes which are assigned each object and subject. The classes are defined with a dominance relationship. An access class c_1 dominates access class c_2 if and only if security level of c_1 is greater or equal to c_2. In order to achieve the confidentiality, two principles formulated by Bell and LaPadula [17] must be satisfied:

No-read-up A subject is allowed a read access to an object only if the access class of the subject dominates the access class of the object.

No-write-down A subject is allowed a write access to an object only if the access class of the subject is dominated by the access class of the object.

These principles ensures that objects cannot be reached by lower level access classes in order to perform a read operation and also objects cannot be modified by subjects which are in higher security level. If a user would like to modify a file which is in a lower class, then (s)he has to connect to the system with a level below its security level of access class [3].

Another important branch of MAC mechanisms is the integrity of the information. Although the confidentiality of the objects could be satisfied with the model above, it does not safeguard that integrity of the resources. For instance, subjects of a low level access class are able to indirectly modify the content of the higher level objects

which threatens the integrity of the resource. As a result, another model for MAC is introduced for this purpose. Biba [18] has come up with the idea for ensuring integrity is maintained. In the Biba model, subjects are not able to change the content of the objects in a non-straightforward way or improper information flows. This model also requires two principles to be satisfied in order to provide integrity:

No-read-down A subject is allowed a read access to an object only if the access class of the object dominates the access class of the subject.

No-write-up A subject is allowed a write access to an object only if the access class of the subject dominates the access class of the object.

By applying these principles, undesired subject access which may cause violation of the integrity of the object is prevented. The principles of Biba model maintains integrity for indirect modification threat whereas integrity itself is much broader concept and additional precautions should be considered [3].

As it can be intuitively figured out so that both models, grants subjects to access to certain direction and the direction is reversed in both models. Therefore, to obtain both confidentiality and integrity as a whole, Bell-LaPadula and Biba models both should be applied to the system. The outcome of the combination of both models is that subject is able to read or write only the objects which are at the same security level as the subject itself [19]. Even though mandatory access control protects indirect information leakages, it is not give the assurance of complete secrecy of the information [3].

2.1.3 Role Based Access Control (RBAC)

Role Based Access Control introduced with the advancements of multi-user and multi-application on-line systems in 1970s [20]. The motivation behind RBAC is simplifying the access control mechanism while maintaining the security policy administration and having flexible access control policies. In this model, system administrators are predefining which roles are able to act according to access policy decisions. Permission determines the actions which can be done when a particular service is requested. Once role-permissions mapping is defined, users assigned abstract attributes “roles” [21]. As a result of this predefined process, it is simple to assign users to roles instead of assign-

ing each user to privileges. RBAC models have evolved so that they are now considered as generalized version of access control. RBAC is general enough to implement both MAC and DAC [20]. On the other hand, RBAC achieves this implementation with less overhead compared to DAC and MAC models. According to Ferralio et al. [22], if U is the number of users and P is the number of permissions in an access control mechanism; the number of administrative operations is proportional to $U \times P$ in identity based authorization and it is proportional to $U + P$ in RBAC assuming the number of roles is constant. The meaning of this is that; if any set of permissions has to be changed for a given role, then only permission-role mapping is going to be modified instead of changing each user permission.

In a project, which is held in the National Institute of Standard and Technology (NIST), it is claimed that RBAC addresses the commercial and governmental requirements such as: user confidence, privacy of personal information, hampering of unauthorized distribution of financial assets [23]. Organizations tend to have access control over users in a centralized fashion, but while maintaining this central approach, they do not want anyone to be able to abuse privileges to any user [20]. Therefore, assigning users to roles rather than the privileges themselves, gives the opportunity to give users predefined set of access over required actions. Eventually, it generates abstract permissions that controls the access rights for a given entity [16]. That is to say, it enables systems to work with abstract data. In addition, it supports for the Principle of Least Privilege [24]. The principle ensures that an entity is only given the permissions to complete a specific operation. As a result, entity has the minimum number of permissions in order to achieve necessary access grant. This principle prevents users to perform unnecessary and potentially harmful action which is a contribution to side effect of granting access to those operations [5].

It is also possible to have a hierarchical ordering between roles. The ordering can be achieved with the introduction of partial order between roles [25]. This order gives ease of assignment of permissions in a well defined fashion. Senior roles may encapsulate junior roles in terms of permissions. With the help of hierarchical rationale, users who share the same level of role can be assigned into a single role abstraction. In other

words, it can classify permissions of roles and enables multiple hierarchies to classify partial order between entities. At the top of the hierarchy, administrators can give partial inheritance between roles under favour of partial order [26]. Figure 3, shows the aforementioned architecture of RBAC.

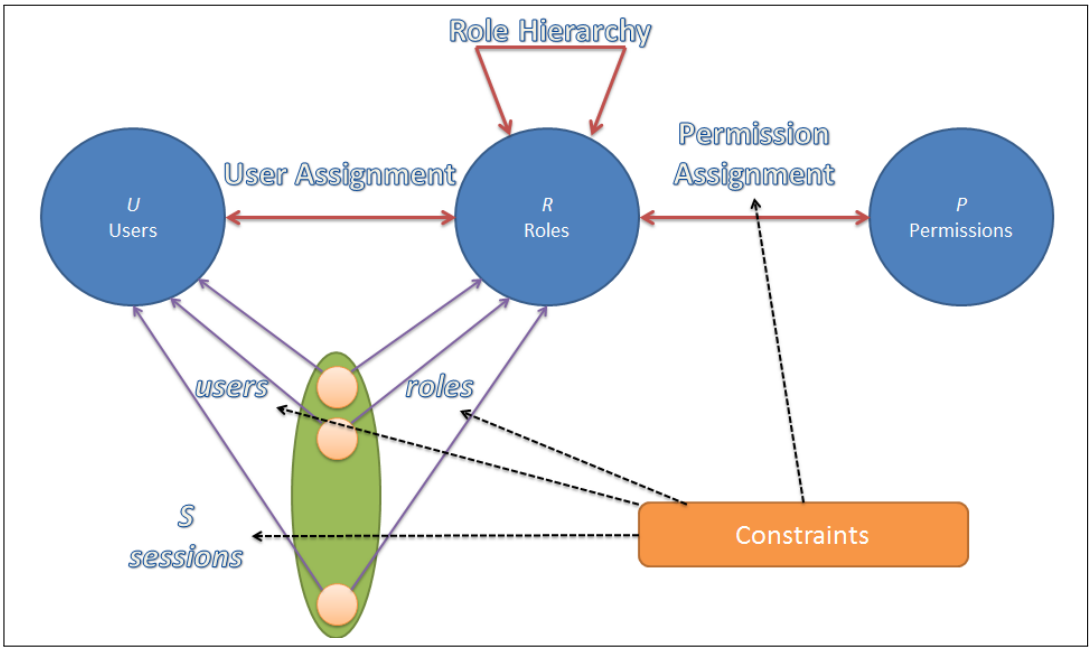


Figure 3: RBAC architecture.

2.1.4 Context-Aware Access Control (CAAC)

Context-aware access control is an extension to RBAC model. It implements the RBAC properties with additional context-based security policies. The definition of *context* is varied in literature [27,28]. In general, it refers to the characterization of physical world situations that are relevant for performing appropriate actions in the computing domain [29]. Contextual information of a subject may be location, the time for access request, computing capabilities, devices being used and such physically related conditions. The requirement for this model comes from the complexity of distributed, heterogeneous domains [30]. The context directly affects the level of trust associated with a user and as a result access is granted or denied for request. The addition of context awareness provides dynamicity for the management of accesses. The trust level shifts according to the context information of the subject.

A generalized version of RBAC (GRBAC) is defined by Covington et al. in order to utilize access control over private information and resources in a ubiquitous computing environment [31]. Environmental roles are included in this model additional to traditional RBAC. Objects are assigned to those environmental roles according to the security policy. The access to the objects are granted if subject satisfies both traditional role conditions and environmental role conditions.

In another model, proposed by Chakraborthy et al., subject can activate a permission and access data in relation to the level of trust has been obtained from the system [21]. The level of trust is calculated for each subject with the help of role and context information. The context information is based on behaviour, knowledge and recommendations by other subjects.

Context information can be also an emergency condition according to the work for criticality-aware access control model [32]. In the work, rather than direct context changes of subjects, the changes of physical environment itself is considered as context. Their claim is traditional CAAC models are reactive and depend on observe/evaluate over the system for explicit access requests. However, those actions does not take into account for emergency conditions. Their work is proactive according to the emergency condition. The condition may be a tornado warning which should automatically tell smart home application in order to unlock doors.

2.2 Privacy of Medical Data and Diagnosis

Privacy has become an significant part of the digital world. Its importance comes from the information it contains. Private data (such as age, birth place etc...) does not seem to have valuable information at all. However, such information may cause unwanted consequences if they are known to third parties. Consider the scenario that an insurance company is going sell health insurance to a person. If the company knows the person had heart attacks in previous years, then the company may exclude the heart diseases from the insurance contract for that particular person. Consequently, insurance companies would start to make contracts according to the health conditions of the people. As a result, companies would tremendously reduce the risk of giving

money to their customers as sudden changes of health condition of customers. That is why people are not and should not be willing to share their private data.

Medical data is a private data of an individual therefore, in this section private information is going to be explained in detail. Also since medical data of individuals are concerned, important medical aspects are also going to be described.

2.2.1 Private Information

The concept of privacy is hard to define. Although it is easy to explain privacy violations, preferences, characteristics and functions, defining the privacy is because its meaning is contingent on culture, situation and personal preferences [33]. One of the famous definition for privacy is defined by Altman [34] : "selective control of access to the self or to one's group". It illustrates private information should not be open to anyone but the predefined set of subjects are able reach the data.

Privacy in medical environment is encapsulated as a multi-dimensional establishment which consists of three independent dimensions: informational, physical and psychological [35]. The first dimension is about the degree of the control over personal information. Physical dimension controls the degree of inaccessibility to others. And the last dimension is the degree of doctor's respect about patient's cultural beliefs inner thoughts, religious choices. Information security mostly concerns about the first dimension as well as the second dimension. Informational privacy is based on person's own decisions over their private data. Individuals would like to have control over their information in a way to determine how, when, where and to what extent the data is going to be shared with another entity. Information security and access control are mainly built on informational privacy. Because it includes avoiding unwanted actions from other entities namely, maintaining unauthorized disclosure from third parties. Information leakage related to patients' health records have caused several reports such as hospital workers were fired because they reviewed it without patient's permission, information related to cancer treatment has shared with National Inquirer caused hospital employees are warned, suspended their work or fired due to the sharing without permission [36].

In pervasive healthcare services maintaining mobility, portability, access authorization, privacy and security is the most important challenge [37]. Through context awareness, a healthcare system can use the context information of the subject to perform tasks according to the predefined physical space. Also the more information flowed to the healthcare system, it can better adapt to serve the user. Paradoxically, the more system knows about the user, it generate a greater threat to the user's privacy [38]. Therefore, maintaining a balance over access authorization and privacy becomes a crucial aspect of a system where private information flow is integrated.

As an example of the challenge, Chan and Perrig [39] worked on the privacy and security over sensor networks. In the work, sensed data through sensors are private data of a patient. They claimed that without ensuring the protection of the privacy of information, it should not be deployed such technology because it will cause more damage than it would otherwise help people.

O'neil at al. [40] worked on personal information security. They investigated commercial framework case studies for electronic commerce system. They come up the use of private information could be put on a beneficial use. On the other hand, it often results in personal information being unwillingly used, sold or otherwise disseminated, and may considered as a form of invasion of customer's privacy. One of the solutions they proposed to overcome the problem is adding anonymity between consumers and institutions. Another solution is the separation of the data over different databases. It can also be illustrated as keeping eggs in different baskets.

In another work [37], balancing usability and privacy while developing security is concerned. Their claim was deployments of pervasive solutions in medicine come up with legal and ethical complications and inappropriate disclosure of medical records involves real and substantial liabilities. Therefore, developing privacy based security systems requires careful considerations of how to comply with legal regulations' privacy and security titles.

All in all, privacy conservation is an important issue in all applications. Developments related with private data should be applied with consideration of issues related with legal regulations. Without taking into account, the consequences of private data

leakage would result in unwanted liabilities.

2.2.2 Vital Signs

Generating diagnoses for a certain illness is an iterative process. This process includes information gathering and hypothesis generation. Data acquisition requires physical examination. This data is crucial for the diagnosis and treatment of the disease. However, each data unit has a potential to change the way of treatment. Diagnostic tests are applied during this data gathering phase. Finding a treatment according to the physiological signs, relevant situations are considered and clinical expert should understand properties of reliability and accuracy as well as the appropriate likelihood ratios. Thus physical examination plays an important role in generating hypothesis about the illness and according to the hypothesis, the treatment which going to be applied is going to be determined.

In the light of this requirement of physical observation, vital signs are the most common examination parameters those are often observed to detect first clues about the disease. There are five vital signs which are considered to be examined first: (i) body temperature, (ii) heart rate (pulse), (iii) respiration, (iv) oxygen saturation and (v) blood pressure [41].

Primal Vital Signs:

Body Temperature is the level of heat produced and sustained by body processes. Variations and changes in body temperature are indicators for possible diseases or other abnormal activities of human body [42]. This sign is important since it affects biological activities of the human body directly. The temperature should be in optimal values for reactions taking place in cells. If temperature becomes higher or lower than the optimal value, actions, which are performed in human body, are going to take more time to complete. If the vital actions are done slower, it would endanger the body due to this slow activity.

Heart Rate or Pulse is the frequency with which the heart beats, calculated by counting the number of QRS complexes per minute [42]. Pulse indicates the speed of heart's blood pumping speed. Therefore, higher pulse is a reflector that heart requires to work more than the expected. The reason behind this overwork may give clues about heart is having trouble with pumping functionality so it beats more or body requires more blood circulation in order to operate functionally. Conversely, if the pulse is weak it could also refer to a problem. The body requires to have a certain flow of blood within the veins in order to continue its biological activities. If it becomes lower than some certain level, it also would threat the life of the patient.

Respiration is the exchange of oxygen and carbon dioxide between atmosphere and body cells [43]. Respiration is significant because it arranges the required external energy resources by taking oxygen and emitting carbon dioxide. Since all biological activities requires energy, respiration directly affects the amount of energy can be generated for body. Low respiration would result in lower energy generation for body. Thus it is going to complete lesser number of vital actions for living. As a result patient's health may go into a state which endangers its life. The opposite way may also cause problems as well. Higher respirations brings about more heart rate per minute. Therefore, it also can be dangerous for sustaining vital activities.

Oxygen saturation is the amount of oxygen bound to hemoglobin in the blood, expressed as a percentage of the maximal binding capacity [44]. Human body needs and regulates oxygen in order to generate energy for body activity. If the balance of oxygen level does not exceed a certain level, than body lacks from the energy to continue body functionality. Although this part is highly related with the respiration, even breathing continues on normal levels, the density of the oxygen may be lower than normal value. Therefore the body triggers frequent breathing to get more oxygen. As it is stated before, lack of oxygen level could result in a dangerous condition. Therefore, this vital information could be used for physical examination of a patient.

Blood Pressure is the pressure of blood on the walls of any blood vessel. It consists of two pressures. Diastolic blood pressure is minimum value of recorded blood pressure. The highest value at which arterial system requires to operate is called systolic blood pressure [45]. When blood pressure is high, it means heart is working harder. It puts extra strain on arteries and heart itself. Over time arteries become thicker and less flexible. This increases the risk of damaging end-organ [46].

As it is depicted, primal vital signs are important to generate a hypothesis about diseases. The data collected from the patient can now put a light on the way of the process understanding the main cause of the illness. Apart from those vitals, there is another important sign that could help the medical personnel to generate hypothesis.

Electrocardiogram or ECG is a graphic record of the heart's integrated action currents obtained with the electrocardiograph displayed as voltage changes over time [47]. In Figure 4, which is copied from [48], the waveforms which consist an ECG record and their intervals can be observed. By monitoring ECG, medical experts would have a clear understanding the causes of an illness. Analysis of ECG record is a crucial element of diagnostics in deteriorating heart diseases [49]. In some cases, the record gives detailed information about non-heart related diseases. This occurs the indirect effects of a disease cause changes in ECG data.

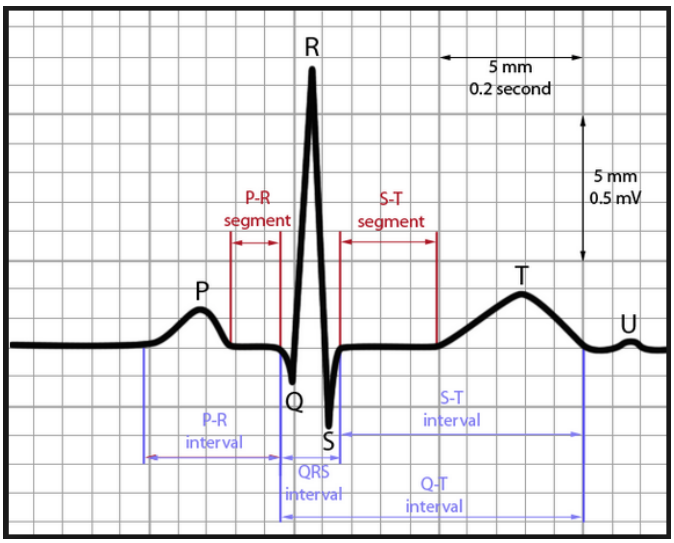


Figure 4: ECG record: Important waveforms and intervals (copied from [48])

2.3 Cryptographic Properties

Security in a computing environment is the protection of digital assets from unintended or unauthorized access. The assets are varied from computer itself to digital information which contain within computer. Security is an indispensable part of this work because as it has been told in Section 2.2.1, the model is established for the private data which is patient information. Therefore, constructing a model with network security becomes mandatory.

2.3.1 Symmetric Key Cryptography

Security has been in use from ancient civilizations. Before 20th century, security concept has been constructed and applied with the symmetric cryptographic systems. Symmetric key cryptography is basically based on a function which takes two parameters one is *cleartext* and the other called as *key*. After the function operation cleartext becomes *ciphertext* which does not directly give any information about the cleartext. In order to retrieve cleartext from the ciphertext, general approach is decipher the ciphertext with another function which operates reverse with respect to encryption function. This inverse function takes ciphertext and key as input and produces decrypted information which is expected to be cleartext itself. This whole method is called symmetric key cryptography because same key has been used for two operations [50].

The most basic example for the symmetric cryptography schemes is Caeser's shift cipher. It's a substitution cipher which replaces each letter in alphabet with another letter. If the letters shifted by 1 to left, then all letters are going to be shifted left by 1. Letter 'b' becomes letter 'a', letter 'a' becomes letter 'z' and so on. To decrypt the encrypted text, applying reverse function as shift right by 1 letter is going to give the cleartext as a result. Another example for the basics of symmetric key encryption is the famous Exclusive or (xor). This is a logical operation that takes bitwise inputs and return true (or 1) if and only if one input is different than the other. XOR is manipulated as both encryption and decryption function. As a result with same key, say 1, if we encrypt 0, we will have ciphertext as 1 and we apply the same operation to decrypt ciphertext 1 to get the plaintext 0.

IBM has conducted a project named LUCIFER which is led by Dr. Horst Feistel [51]. At the and of this research project, an encryption algorithm for data protection was published. This algorithm is based on the feistel network which ensures that there exists an inverse function. The algorithm later become a standard for symmetric key encryption named as Data encryption standard (DES). It originally takes 64-bit key, input and output. However, the implementation does not use all of the key namely 8-bit of the key does not used during encryption. Those bits are called as parity bits. In short the algorithm utilizes 56-bit key. It was a strong algorithm during 1970s since computing power was much more less than today's.

56-bit key is considered as not secure once computing power increased during the years. The first attempt not to change the standard but increase the security was the invention of 3-DES. It increased key size from 56-bit to 168-bit if three different keys are used. 3-DES can be also used with 2 different keys then it will have the security level of 112-bit. 3 different DES keys are used to generate ciphertext. However, with its vulnerability to meet-in-the-middle attack [52], efficient key-size becomes as same as 2 different DES keys which is 112-bit.

112-bit efficient key-size become less secure due to the advancements of computing power. A new standard is required to be established for data security. Vincent Rijmen and Joan Daemen has won the competition which was organized by NIST [53]. Their work on symmetric encryption become the standard for data encryption. It is approved by National Security Agency (NSA). This cryptosystem can be used with three different key sizes; 128-bit, 192-bit and 256-bit. The name of the encryption scheme is Rijndael but it is generally known as Advanced Encryption Standard (AES). The attacks found on AES system still require computational complexity which are close to exponent of the key-size. Therefore, AES is still applicable for today's computer security requirements.

There is a drawback with the use of symmetric key cryptography which is called key distribution problem. As it is mentioned both encryption and decryption require to have the same key in order to have a proper communication. But distribution and management of those keys are problematic due to initial communication to agree upon a key in a public network.

2.3.2 Public Key Cryptography

Public key cryptography (PKC) has been first introduced at 1970s by Withfield Diffie and Martin E. Hellman [54]. They step in to the problem of key exchange and give the notion of digital signatures. The cryptosystems based on public key cryptography can be proven to be secure because they require computationally too much time to break the system which is considered as infeasible. Public key cryptosystems are mainly built on three of the big number theory subjects; *Discrete Logarithm Problem* [55], *Integer Factorization* and *Elliptic Curve Cryptography* [56]. In PKC, function that encrypts plaintext takes key parameter which is called as *public key*. On the other hand, function which decrypts the ciphertext takes key parameter as *private key*. These two keys are different from each other but they are not completely independent from each. Since reason two functions use different keys, PKC is also termed *asymmetric key cryptography*. Public key algorithms are less efficient with respect to symmetric ones because they generally require more computing operations therefore they need more time to complete computation. [50].

Diffie-Hellman protocol [54] was the first attempt to solve key exchange problem over a public channel. Figure 5 describes the notion of the key exchange for two parties. Basically two parties first agree on a multiplicative group of integer under modular of a prime number and also select a generator of this group. Then, both entities chose their secret number under this multiplicative group. Both parties send to other side the exponent of the generator with the secret number under modulo of the agreed prime number. The security comes from the discrete logarithm problem. It is computationally infeasible to find secret from computation result of modular exponentiation for big numbers. Even though protocol is a novel one, it is vulnerable to man-in-the-middle attack [57].

Rivest, Shamir and Adleman published the RSA [58] algorithm as another public key cryptosystem. In this system, it is possible to do both encryption/decryption and digital signatures. RSA is based on the integer factorization problem. Figure 6 gives the computations of key establishment phase of RSA. One selects two big primes and multiplies them to get a bigger composite number. This composite number is used

1-	All sides agree on a $p \in \boldsymbol{P}$ and generator g under multiplicative group of p	
	A	B
	---	---
2-	Select a secret α in the group	Select a secret β in the group
3-	Compute $g^{a}(mod\ p)$ and send to B	Compute $g^{\beta}(mod\ p)$ and send to A
4-	Compute $(g^{\beta})^{a}\ (mod\ p)$	Compute $(g^{a})^{\beta}\ (mod\ p)$
	$(g^{\beta})^{a} \equiv (g^{a})^{\beta}\ (mod\ p)$ so both sides agree on a key	

Figure 5: Diffie-Hellman Key Exchange

for both encryption/decryption and digital signature operations. Before doing cryptographic operations, another calculation must be made. This is finding the number of relatively prime integers which are less than or equal to composite number. Finding those numbers also referred as calculating *Euler's phi function* (Φ) [59]. As a final operation, public and private values of RSA are determined such that both of them should be relatively prime to the result of Euler's phi function. Encryption and signature validation is done with the help of public key. Public key is conceptionally is not trusted by everyone but since it is public anyone can do encryption and signature validation with the information. On the other hand, decryption and generating signature from a plaintext is only available with private key. The private key should be the secret key which is known solely by its owner therefore owner becomes the only entity which is able to sign files and able to decrypt incoming messages which all operated with that particular public key.

1. Select two large primes $p, q \in \boldsymbol{P}$
2. Calculate $n = p \cdot q$
3. Calculate $\Phi(n) = \Phi(p \cdot q) = \Phi(p) \cdot \Phi(q) = (p-1) \cdot (q-1)$
4. Select a random integer e such that $0 < e < \Phi(n)$ where $gcd(e, \Phi(n)) = 1$
5. Calculate $d \equiv e^{-1}(mod\ n)$ where $0 \leq d < n$

Public Key: (e, n)

Private Key: (d, p, q)

Figure 6: RSA Key Establishment

3 Related Work and Problem Statement

In this section, related work in the literature is discussed. Problem statement of this work is also mentioned in this section.

3.1 Related Work

Today's access control models mainly use RBAC principles in order to reduce the number of control operations over a target subject. Zheng et al. [60] defines participation, act and activity in order to obtain a dynamic version of RBAC. Act is defined as an operation of application systems and role is defined as a set of subjects sharing the same access control policies to certain objects. Participation denotes a functional role and co-works with act; it is a new abstraction between roles and acts. First, the role of a subject that requests access is found within the system. Then according to that role, subject is granted participation controlled by defined rules in access control policy. If participation of a subject is mapped to requested act in activity cell, then access is granted to subject.

A RBAC mechanism is also constructed for cyber-physical systems by Muppavarapu and Chung [1]. They try to reduce the administration overhead, which stems from the role privileges of the individuals by a middleware. They apply a protocol to gain access control credentials and once those credentials are obtained, the protocol communicates with the resource manager in order to perform the requested operation.

The abovementioned two studies [1, 60] do not address the criticality management requirement of our proposed model.

Venkatasubramanian [2] claims that in a medical environment access control should be adaptive, and therefore, dynamic for emergency management. This versatility pro-

vides the required privileges to the subjects implicitly for short periods of time. With the use of critical-aware access control, a model has been constructed, which behaves like context based access control (CBAC) in normal state. In CBAC, context information of the subject determines the access control. For instance, context can be constrained by time and space. If a subject requests access in different places at the same time, system rejects requests according to the policy of having a subject not to appear in different places at the same time. Other than normal states, when someone experience criticality, it shifts from this model to another, which is more proactive in nature.

Undoubtedly, the work proposed in [2] is closely related to our study since it supports criticality management. However, it achieves the regulation of critical situations by applying regular checks over the system in certain periods. Another drawback of [2] is that it tries to automate the responsive actions over patients for a calculated amount of time. This is a medical risk, because treatments cannot be applied to all patients in the same way even if they suffer from the same disease. Therefore, we come up with a model which interprets patients' medical information whenever the data are received by system. Under critical circumstances, system dynamically gives extra control to medical professionals in order to recover the patients from their critical diseases.

3.2 Problem Statement

Access control has been an important topic where selective restriction is required for certain resources. It is actually a process based on prevention of unauthorized use of a resource [61]. Most of access control models rely on authorizing identity of the user and directly inspects whether that user is eligible to have the requested information. In a medical context, the information retrieval becomes more crucial due to the access over data could affect response time of a emergency condition of a patient. Thus, access control model should prevent unauthorized accesses and also it should respond to the requests in a short period of time. Because of these reasons, we aim to bring three important properties to access control over medical data. The first one is dynamical change of access policies due to emergency conditions. The second one is the real-time interpretation and analysis of medical data. As a third property, system gives subjects

the opportunity of having more than one access right at a given time.

- **Dynamical change of access policies:** Under emergency conditions of a patient, the access to private data requires some flexibility for the sake of quicker medical response. Moreover, saving patient from such emergency conditions may rely on getting help from more than one medical expert. In order to receive this help, those medical experts should observe the condition of the patient by requesting access to his/her medical data. Consequently, system should dynamically change access policies to deliver medical information to medical experts.

- **Real-time interpretation and analysis of medical data:** In pervasive healthcare systems, patient data is sensed and transmitted over a network to hospital server. In this server, doctors are able to monitor the health conditions of the patients. For the sake of understanding the medical condition of the patient, interpretation and analysis of medical data are required. Moreover, it should be considered if the patient is experiencing an emergency condition, then this should trigger an alarming state. This is essential since doctors may not be aware of the condition at the time criticality occurs. Also, the situation is often needed to be responded promptly. Correspondingly, the interpretation and analysis of the medical data are necessary to be done in real-time. It is going to directly affect the health condition of the patient. In most of the cases, a timely intervention increases the chance of prevention of deterioration and/or complications [62–64].

- **Maintaining multiple access right at a given time:** General approach in access control systems is controlling whether requested access is a valid one with respect to the defined rules. In RBAC, subjects have predefined roles over the access control manager. Therefore, their capabilities are controlled by their roles. In our system, system users are able to have more than one role to get multiple access rights at a time. Consider a scenario that a medical expert also requires to use our system as a patient. The scenario could be established in an opposite way; a patient defined in the system may become a medical expert as well. If the access rights are defined properly, the risk of giving permission to an unauthorised

subject is eliminated. Moreover, system is able to give permission to subjects with respect to their multiple role access requests at the same time.

As an outcome, the ultimate goal of this thesis is construction of an access control model for medical information which also have dynamical properties as a response to emergency conditions and able to interpret and analyse the medical data in real-time. In the following sections, we explain our methods and protocols how to achieve this goal. Moreover, we also provide simulation-based performance evaluation results.

4 Proposed MAR-BAC (Medically Adaptive Role Based Access Control) model for healthcare systems

Healthcare systems are used to generate and transmit medical records from the source to a sink, which collects data from distinct subjects. During this transmission, medical data is open to unauthorised accesses and modifications if the network is not secured. Even in secure and private networks, the integrity of the data may not be maintained because of the transmission errors. In healthcare management, public communication channels are generally used. Therefore, it is going to be open for inner and outer threats in terms of privacy and access permissions. To overcome such problems, we propose an access control model which prohibits unauthorised actions by applying additional security checks specifically for medical environments. Access control policies are dynamically adapted while ensuring the protection of the digital data. Since we deal with medical records, we are able to analyse the medical data and interpret the health condition of the user. This is important because the health status of the patient may change in a negative way. There could be a situation which requires an external help in order to recover from the problematic condition. Those conditions are called critical conditions or criticalities. The anomalies found in the patient's medical record are recognised by the system automatically and the information is used to notify medical personnel to cooperatively rescue the patient's life. Our system utilises parts of RBAC and CAAC models for the application of a healthcare system. It has been constructed such that roles are valid in a certain period of time. Another benefit of our system is that access control constraints and policies are defined according to the

needs of the healthcare systems. As our system manages multiple users, actions should be clearly defined for each role and user. In short, access control is required in order to manage the operations in an order. In our work, an access control model has been described which is composed of different phases. Before going into detail, the definition of the identifiers which are used in this model are given in Table 1.

Table 1: List of identifiers used in MAR-BAC Mechanism

A	Set of Administrators	K_{Server}	Public Key of ATOS
$ADPS$	Authorization and Data Processing Server	M	Set of Medical Experts
$ATOS$	Authentication and Ticket Obtainment Server	OTP	One-Time Pad
APM	Access Policy Manager	P	Set of Patients
C	Set of Disease Category	P_j	Assigned set of patients to medical expert $m_j \in M$
CP	Set of Control Policies	$PU(key, plain)$	Public key encryption of plain using given key
D	Set of Diseases	R	Set of roles
$D_{i,t}$	Set of possible diseases for patient p_i at time t	T	Set of Time
$E(key, plain)$	Symmetric key encryption of plain using given key	$Ticket_{Id_{ADPS}}$	Ticket assigned to server identifier
H	Set of Health Information	TS	Time Stamp
Id_{ADPS}	Identity of ADPS	U	Set of Users (subjects)
K_{AA}	Pre-shared key between ATOS and ADPS	α	Set of acts
K_{ATOS}	Key which is only known by ATOS	Γ_i	Access request from user $u_i \in U$
K_{CS}	Shared Key between Client and ADPS	π	Set of participations

4.1 Set Definitions

In this subsection, set definitions, required constraints for access control model and also control policies of the system are going to be explained. This work is an extension of RBAC therefore the roles of the system are defined as follows.

As it is stated in Table 1, U is the set of users, A is the set of system administrators, M is the set of medical experts and P is the set of patients. New roles can also be added in case of need. Currently those three main roles are sufficient to configure the system. Proposition 1 defines U is superset of specific user roles. In other words, set of system administrators (A), set of medical experts (M) and set of patients (P) are subsets of the user set (U).

Proposition 1. $A \subseteq U$, $M \subseteq U$, $P \subseteq U$.

In healthcare systems, general idea is the transmission of medical data from patient to another digital entity. The medical data to be transmitted can be specified either by the user or system has default options about medical data transmission. Proposition 2 defines a system control policy for patients. In this definition, medical data can be only obtained from a patient and it is system's responsibility to manage medical information of the patient.

Proposition 2. $\forall p_i \in P$, system is responsible for monitoring health information of patient p_i.

The medical information gathered from patients is kept in hospital server. Medical experts defined within the system are able to monitor those medical information under the regulations of the hospital. In this system, we prefer to assign a set of patients to a particular medical expert. This set of assigned patients can be reached with the function given in Proposition 3. With this patient medical expert assignment, medical experts are able to monitor predefined set of patients under the hospital regulations. The information transmitted is the medical data, which is private information of the patient.

Proposition 3. Let $P_j \subseteq P$ be the set of patients assigned to medical expert $m_j \in M$. $\forall p_i \in P$, $\forall m_j \in M$ system reveals information of p_i to m_j if and only if $p \in Pj$

Assignments of patients among medical experts are managed by the system administrators. This control is not given to medical experts, because in such a case, the experts become capable of assigning all patients to themselves. Therefore, it may end up with monitoring whole patients' medical data. This is a potential privacy breach. System administrators have the control of assigning and removing users for certain roles and this assignment can be achieved for certain periods of time. Time constraint is necessary, because a medical expert may required to be defined to system temporarily. If that is the case, defining medical expert without time constraint may cause data leakage problems. However, if the medical expert is able to connect to system for a predefined period of time, this risk is eliminated. Administrators are also able to define mapping between medical expert and patients in order medical experts to monitor patients' health conditions. Proposition 4 defines the administrator capabilities.

Proposition 4. $\forall a_k \in A$, a_k is responsible for updating sets p_j, P and M.

As mentioned before, medical experts are not able to assign patients. With the similar idea, system administrators cannot be able to assign themselves as medical experts at the same time. This is crucial since obtaining a medical expert role provides the opportunity of monitoring medical information. Then the solution for this requirement is exclusion of roles from each other. Proposition 5 illustrates the idea more formally so that a user in medical expert set cannot be an administrator and vice versa. As a result, the intersection of medical experts' set (M) and set of administrators (A) yield in an empty set. Without this control policy, system has the risk of leaking private information of patients. However, a system administrator or a medical expert can be patient, because patients are only be able to request their own medical information, which is a valid request for the system.

Proposition 5. $M \cap A = \emptyset$.

The aforementioned patient assignment is specified to many-to-one relation. A medical expert have multiple patients defined within the system, but a patient can be assigned only to a single medical expert. Proposition 6 introduces the idea of this many-to-one relation from the medical expert point of view. Given two different medical

experts defined in the system, they do not share any patient which is assigned to both medical experts. Consequently, the sets of assigned patients (Pj) are partitions of the patient set (P).

Proposition 6. $\{P_j | m_j \in M\}$ is a partition of P.

Medical information for a particular patient expectedly varies from time to time. In order to specify the health condition of a patient at a given time, a function is defined in Proposition 7. This function takes two inputs as patient and time variables. It outputs the health condition of that patient at the given time interval. The experts are able to get medical information of a patient with a given time with this functionality of the system. Also patients can benefit from this function so that they can also monitor their health condition of their own. Under normal conditions, assigned medical expert is the only personnel who is eligible to retrieve medical information of the patient. However, if a patient experience a condition which requires additional cautions to prevent a dangerous outcome, the system should adapt itself according to the condition.

Proposition 7. For a given time $t \in T$, we define the function θ such that $\theta(p_i, t) = h_{i,t}$, where $p_i \in P$ and $h_{i,t} \in H$ at time t.

Patients of the system are able to generate the medical data and send the data in a secure way to the hospital servers. The security of the data during transmission is going to be explained in Section 4.2. From the access control point of view, medical data of a patient can only be accessed by a single user from the patient set P which is the owner of the medical data. Proposition 8 defines this constraint in a way that only the owner of the medical data from the patient set is a valid user for obtaining the access right.

Proposition 8. Let $p_i, p_k \in P$. p_i is able to call function $\theta(p_k, t)$ if and only if $i = k$.

Definitions and functions given up to here basically constitute the access control model which regulates the system under *normal conditions*. Here *normal conditions* means that the patient's health conditions do not yield a criticality after analysis and interpretation of the medical record. Normally, received medical information sent from

patient is logged into hospital server and interpreted by the system in an automatic way. If current condition of the patient requires a medical intervention, system autonomously takes an action accordingly. In such conditions, system shift from normal conditions to emergency conditions for particular patient. For this reason, the critical diseases are defined within the system. After interpretation, data is analysed whether it contains any vital information that is necessary to be responded by a medical expert. Therefore, system initially needs a function so that it can analyse medical information and come up with a list of possible diseases with the given medical data. Proposition 9 introduces a function which takes health information of patient and returns the list of possible diseases if exists. If the medical condition of the patient does not need any urgent intervention, then function returns an empty set.

Proposition 9. Let $p_i \in P$, $t \in T$ and $\theta(p_i, t) = h_{i,t} \in H$. Define a function f such that $f(h_{i,t}) = D_{i,t} \subseteq D$. If p_i experiences a fatal disease, then $D_{i,t} \neq \emptyset$.

The set D consists of different diseases but each of them belongs to a certain disease category. The disease category is an abstract group for diseases. The reason behind this grouping mechanism is the need for the selection of medical experts to be notified when urgent response for medical condition is required. In order to construct a generalization for diseases, and retrieve the category of a particular disease, Proposition 10 defines a function:

Proposition 10. $\forall d_k \in D$, function $\gamma(d_k) = c$ implements d_k belongs to the disease category $c \in C$

With the same idea, medical experts are required to be have specialisation for certain disease category for emergency conditions. To achieve this requirement, Proposition 11 defines a function which takes a medical expert as an input and it returns the speciality of the medical expert. The category of a certain disease and speciality of a medical expert is needed to be predefined within the system in order to operate normally. This function also gives the users the opportunity to assign patients to medical experts who are specialized with the disease category.

Proposition 11. $\forall m_j \in M$, define function $\eta(m_j) = c$ gives the specialisation $c \in C$ of the medical expert m_j.

Once diseases are categorised and medical experts' specializations are defined, finding whether a medical expert is specialized with a certain disease or not becomes only a function check. In Proposition 12, previously defined functions are used to make a control check whether a medical expert is capable to treat a particular disease or not.

Proposition 12. If $\gamma(d_k) = \eta(m_j)$, then $m_j \in M$ is said to be specialized for the disease $d_k \in D$

In case of an emergency, a subject may override the access rights of its role. The solution is not preferred because the notion of emergency cannot be well-defined and is open to abuse by the subjects. However, the abuse is prevented by a mapping function that picks a subject according to the critical data of the patient under risk.

4.2 Protocols for Secure Login

In this work, before establishing a communication between hospital server and user, both entities need to generate a secret key in order to initiate a secure link. Once client and server establish a secure communication channel, user's requests should be controlled by the system whether the request is legitimate to be applied by the server or not. In the rest of this section, the details of establishment of secure channel and process of controlling access are described.

4.2.1 Authentication and ticket generation

Healthcare systems should possess high level of data security since they process private medical data. To ensure the security, our protocol requires a trusted third party, which is called Authentication and Ticket Obtainment Server (ATOS). Our login protocol is a ticket-based one as in Kerberos [65]. Clients who would like to login to the hospital servers, named Authorization and Data Processing Server (ADPS), should first identify themselves to ATOS. These two servers together with clients construct the general overview of client-server architecture which can be seen in Figure 7.

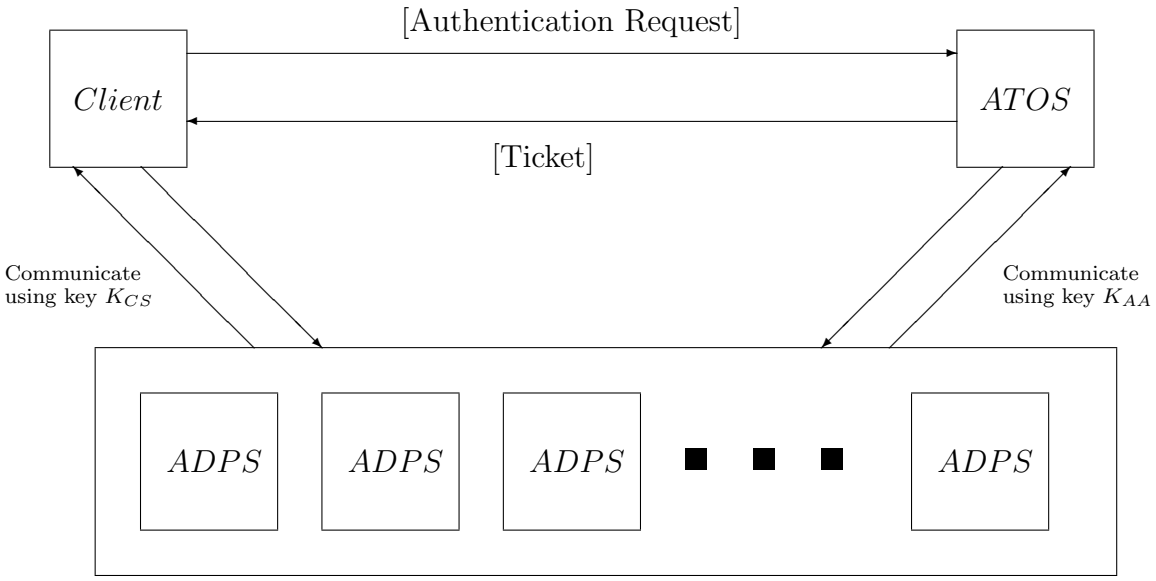


Figure 7: General Overview of Client-Server Architecture

Before explaining login in detail, we give the assumptions for the system.

- Client knows the correct public key of ATOS.
- ATOS has secure communication between all hospital servers (ADPSs).
- Client identifiers and client's pin codes are kept in ATOS database in a secure way.

Authentication and ticket generation can be seen in Figure 8. In step 1, client sends its identity to ATOS together with a nonce in an encrypted form. This encryption employs a public key cryptosystem such that the context is encrypted using the public key of ATOS. This public key is to be known by the client. Nonce is a random string which has the same length with the user's id as they are going to be XORed to generate the session key. The session key is the result of a hash operation which takes XOR of client identity and the nonce value as input. In step 2, ATOS authenticates itself to client by sending a signed version of generated key (K_{CS}). Since client has the public key of ATOS (K_{Server}), the signature can be verified. In step 3, client sends

its pin to authenticated ATOS in encrypted form to complete mutual authentication. While sending pin, client also sends the hospital server identifier (Id_{ADPS}) which (s)he

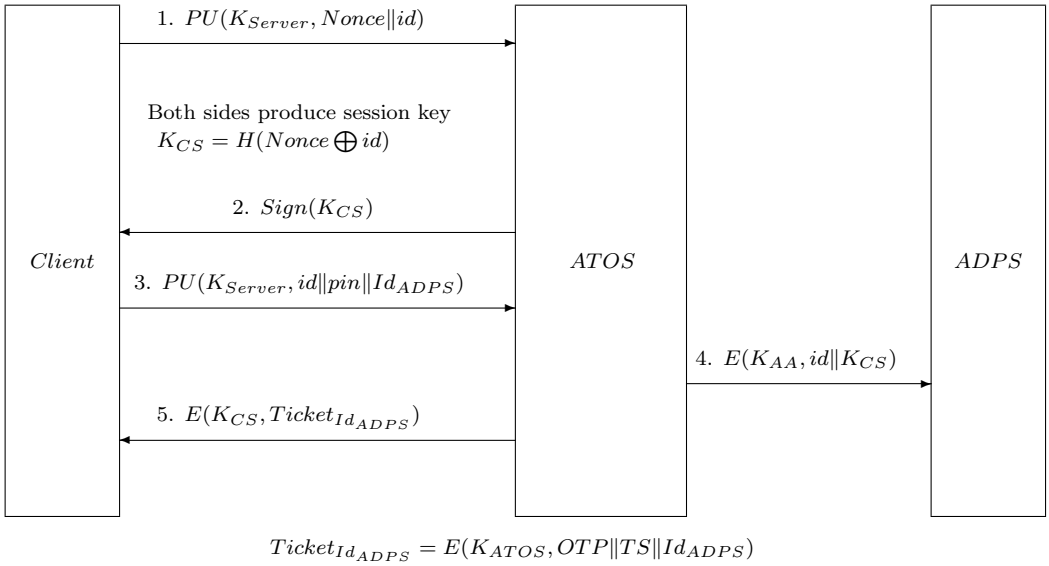


Figure 8: Authentication and ticket generation protocol

would like to login. In step 4, ATOS sends the generated session key to the designated ADPS securely. ADPS and ATOS has predefined shared secret key (K_{AA}) to maintain the secure channel in between. In step 5, ATOS generates and sends a ticket for the communicating client. This ticket contains information about the identifier of the ADPS, to which the client requested to login. The ticket is going to be checked again by ATOS in the ticket validation phase explained in Section 4.2.2. Generated ticket is encrypted with the session key (K_{CS}) using symmetric encryption and sent to client. Tickets cannot be modified by an unauthorized user since its content is encrypted by a key (K_{ATOS}) known only to ATOS. As a result, ticket information is first encrypted with a key which is only known by ATOS then it is again encrypted which can only be decrypted by the client, ATOS and requested ADPS. Ticket also contains a timestamp which is designed to hinder the replay attacks on the system.

If the client successfully completes authentication and ticket generation protocol, generated ticket is going to be used the next phase, ticket validation.

4.2.2 Ticket validation

In the previous subsection, we explained how a ticket has generated by ATOS in order to control client login to the hospital server (ADPS). When the user gets encrypted ticket, (s)he first decrypts it. As mention previously, the ticket itself is an encrypted value which can be validated only by ATOS since it is encrypted using a key (K_{ATOS}) known to ATOS only. This is performed in the ticket validation protocol mentioned below.

The protocol is given in Figure 9. First two steps are summary of the authenti-

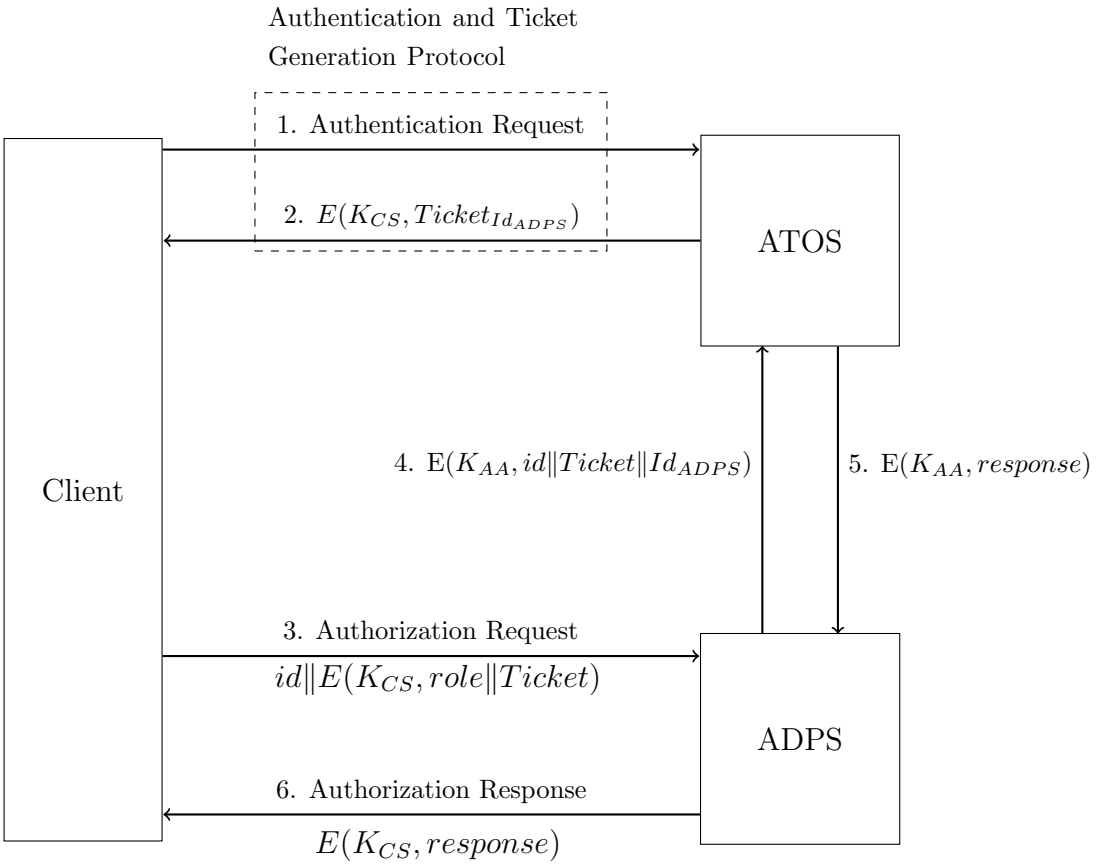


Figure 9: Ticket Validation

cation and ticket generation protocol explained in the previous subsection. In step 3, client encrypts its role and ticket with the session key (K_{CS}). Then client sends this encrypted value together with its identifier (id) to the designated ADPS. In the previously described authentication and ticket generation protocol, ATOS sent the session

key (K_{CS}) of the client to ADPS. Therefore, ADPS is able to decrypt the encrypted value sent by client. In step 4, ADPS sends an encrypted message to ATOS. This message is the encrypted form of client identifier, client's ticket and identity of server itself with a key (K_{AA}), which is shared before with ATOS. After receiving this message, ATOS decrypts it with K_{AA} and obtains the ticket. Then ATOS checks the validity of the ticket by decrypting it with K_{ATOS}. If the thicket is valid, then ATOS generates a positive login response. Otherwise, ATOS generates a negative login response. This response is sent to ADPS in step 5 again in encrypted form. In step 6, ADPS sends the login response to client. If the ticket is validated, ADPS checks the identity of the client and its role within the system. Generated session key (K_{CS}) is used for the confidentiality and integrity of the private data. If a valid user does not send its role during the login protocol, that client have limited access over resources. In this case, the client is able to perform certain operations for a restricted amount of time.

4.3 Access Operations and Access Control Architecture

In this subsection, we explain the access control model and its architecture. Moreover, it is also described how access operations are performed.

4.3.1 Access request and response architecture

In an access control model, access permissions of subjects are required to be predefined. As a result of this predefinition, access responses can be generated in response to the access requests of subjects. Medical data transmission steps from ADPS to client is shown in Figure 10. Client requests information from ADPS as the initial step. ADPS directs this request to the Access Policy Manager (APM). APM is the entity that controls the rules which are going to be applied for MAR-BAC model. The rules and other control definitions are stored in *Constraint & Control Policies Database*. The requested action is controlled whether parameters for the request are valid or not. According to the rules, a response is given and APM forwards this response to ADPS. If the response is positive, then ADPS queries about requested action. The action can be anything defined in the MAR-BAC model. Depending on the access response and query result,

an informative message is sent to client.

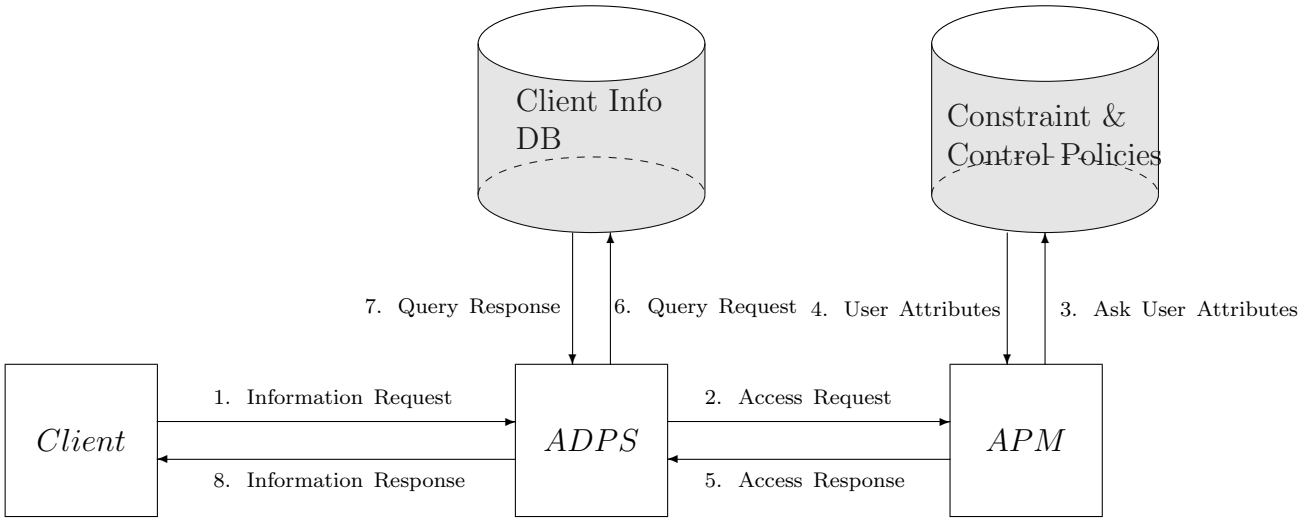


Figure 10: Information Flow from ADPS to Client

The information flow is actually bidirectional; we explained ADPS to clients information flow above. Figure 11 shows the medical information transmission from client to ADPS. Information flow from client to ADPS again starts with a request step. Then

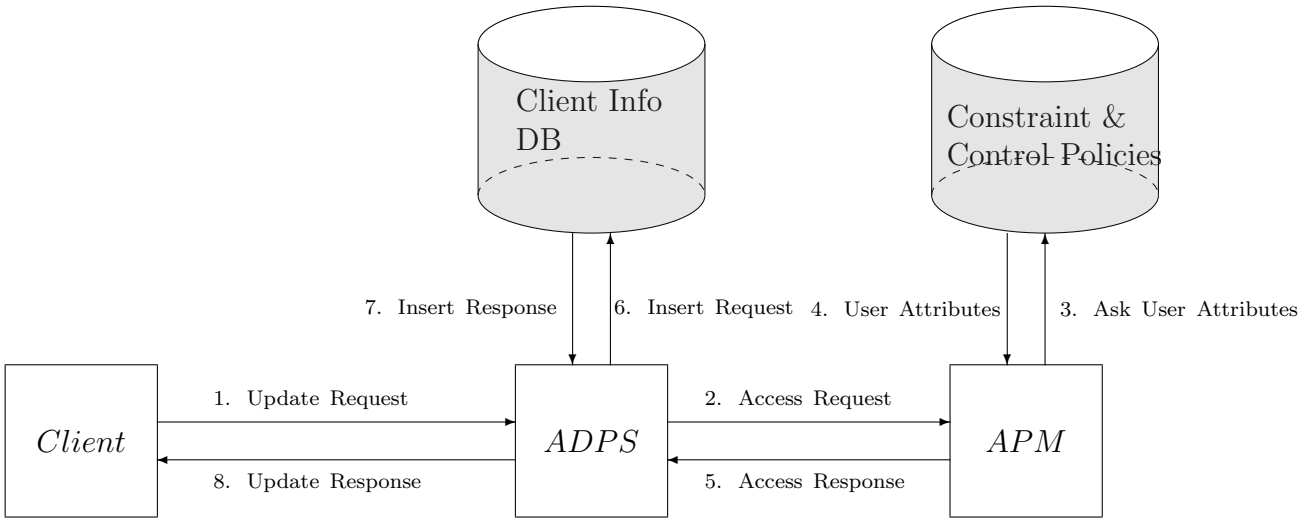


Figure 11: Information Flow from Client to ADPS

ADPS forwards this request to APM as an access request. APM decides whether the subject has the right for inserting/updating information on hospital database. Once access is granted by APM, ADPS inserts/updates the client information database according to the information client sends to the server. Finally an informative message is returned to client as an update response.

4.3.2 Access Control in MAR-BAC

In this subsection, access control model for MAR-BAC, where is shown in Figure 12, is explained. Roles (R) defined in MAR-BAC are mapped into participations (π). Here, the roles are more likely static groups and the participations provide the dynamicity of the access control mechanism. The reason behind this is that access can be obtained by the users when they participate in an act. The act set (α) is defined according to the control policy (CP). The assignment between roles and participations are regulated by the role-participation maps (RP). Participations and acts are mapped with each other within a directed graph called the activity cells (AC). The mapping defines which participation instances are able to do predefined acts.

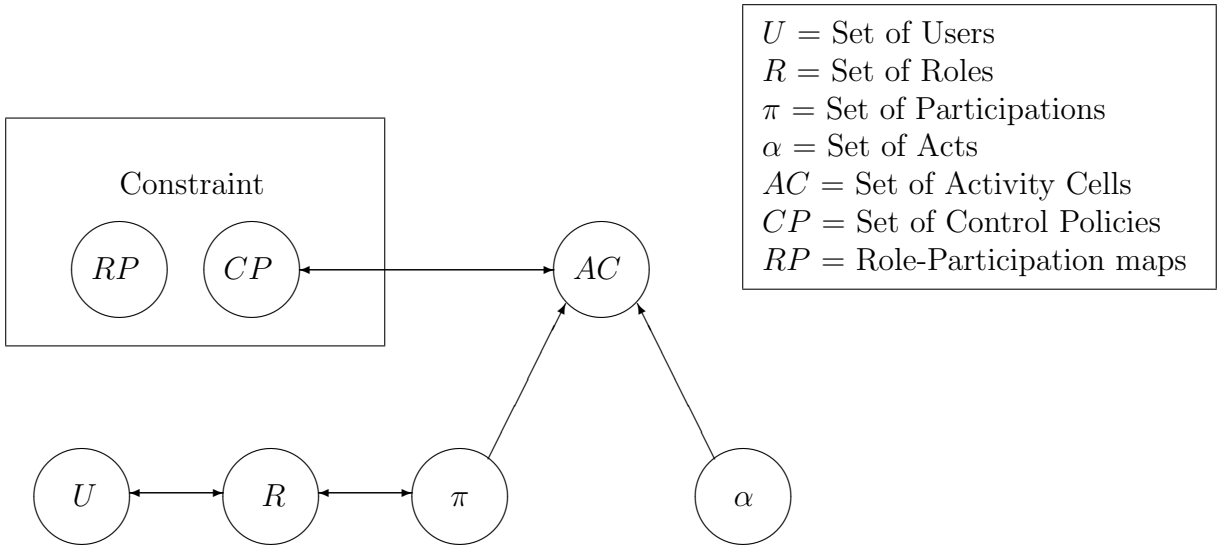


Figure 12: MAR-BAC model

RBAC model does not aware of time. It only controls access policies according to

which role can perform which actions. Therefore, access control mechanism does not take into account time as contextual information. In MAR-BAC model, we extend the RBAC by assigning users to roles for a fixed amount of time. As a result, access control policies are now able to consider context information of time while making decisions about access permissions.

Roles should be defined in RBAC to control the access operations. However, there may be cases such that users are defined to system but they do not have assigned to roles. In such conditions users, who do not belong to a certain role, do not able to perform operations. As a unique feature of our MAR-BAC model, users defined in system without role are also able to perform operations. Since MAR-BAC model aware of time as context information, those users can perform operations under a very strict time constraint.

A user, who does not have any role defined as a patient, is able to send its medical information to ADPS. Since s/he is not defined as a patient, no medical expert is assigned to that particular user. As a result, no medical expert is able to retrieve that users medical data under normal conditions. Such users are able to perform operations as a patient for a given period of time. Meaning that, those users should obtain a role in order to continue as legitimate users. The users, who do not have any role assigned, are able to get access with the assistance of participation set defined in MAR-BAC model.

The access request steps of MAR-BAC are summarised in Algorithm 1. At first, user gets authenticated via login phase as explained in Section 4.2.1 and Section 4.2.2. Once the system controls the validity of the client, server sends access request to APM. In APM, the initial operation for access request is retrieval of the participation list according to the role of the user. Whether client has a role or not, (s)he needs to have a list of participations in order to request a valid action within the system. If a user has a role, APM returns list of participations of that particular role. Otherwise it will get the restricted operations defined in the system.

Participations consist of both functional role and validity time period of a particular user. As it is mentioned before, users are able to perform actions within a predefined

Algorithm 1 Access Request Steps

Require: access request Γ_i from user $u_i \in U$ and role $r_i \in R$ of user u_i

```
if u_i is authenticated then
    π_i = GetParticipations(u_i, r_i)
    requestResponse = CheckActivityCell(Γ_i, π_i)
    if requestResponse is valid then
     return access grant
    else
     return access reject
    end if
end if
```

period of time for each login session. Once APM retrieves participations, it sends the list and requested act to activity cell. Activity cell is a directed graph which maps participations to acts. In activity cell, if requested access right has been mapped in the corresponding act, then APM grants the access; otherwise, it rejects the access request.

Once access request is granted by APM, system performs the requested operation and sends an informative message to user. If the access request is a medical information transmission from patient to ADPS, then a medical analysis is performed by the system.

4.4 Medical Analysis

In this part we explain how collected medical data from the patient is interpreted and analysed. Physiological signs play an important role in the diagnosis of diseases because those signs assist medical experts to generate hypothesis about the illness. In MAR-BAC, we utilise the interpretation of some of the primal vital signs and ECG signal retrieved from patient.

Our system is initiated whenever medical data is retrieved from the patient. Once the data of the patient has been securely received by the hospital server (ADPS), the data is first stored in a secure database. Then the data is sent to the medical data interpreter subsystem. In the medical data interpreter, the primal vital signs (body temperature, blood pressure, respiration, oxygen saturation and pulse) and ECG signal are analysed and interpreted.

Most of the biological activities are directly affected by the body temperature.

Therefore, interpreter initially checks for the body temperature of the patient. Temperature higher than 37.5 ℃ is considered as high. Since those values may critically malfunction the body activity. Therefore, patient needs immediate responses in order to recover its health. Similarly temperatures lower than 36.0 ℃ is interpreted as low body temperature in our model. Values between 37.5 ℃ and 36.0 ℃ is optimal value for body temperature.

Another determinant factor for the health condition is the blood pressure. It consists of two different types; diastolic and systolic. As the second phase of interpretation, patient's blood pressure is observed by the system. If the blood pressure is below 90 mmHg for systolic or below 60 mmHg for diastolic, then the interpretation result is *low* for the patient. If systolic value is higher than 150 mmHg or diastolic value is higher than 95 mmHg, then the result is *high* blood pressure.

The next phase of the interpretation is inspection of respiration. It is another significant factor due to breathing is a requisite for human life and respiration disorders could harm the body. If the system finds out that the patient's respiration frequency is more than 26 per minute, then system interprets the result as high respiration frequency. Respiration rate below 14 breathes per minute is interpreted as low. With low rate, patient's health condition can be severely damaged since the body lacks energy. Values between 14 and 26 per minute are considered as normal values for respiration frequency.

In addition to respiration frequency, it is also significant how much oxygen is saturated by blood cells, as it is another significant factor of energy generation. Thus, oxygen saturation is another parameter to be checked by the system. Oxygen saturation is not sufficient if it is below 90%. Otherwise it is interpreted as normal.

As a last primal vital sign, pulse is analysed by the system. It is another important characteristic that medical experts check under emergency conditions. It directly affects health condition because heart beat is one of the vital operation that human body has to continue for survival. Upper bound for heart beat rate is 100 beats per minute. Lower bound for the frequency of the pulse is 60 beats per minute.

All of the primal vital signs have the potential of pointing a critical disease which requires timely actions by the medical experts. Although primal vital signs are ini-

tial parameters to be examined, in this work electrocardiogram (ECG) information is also checked in order to make more precise decisions about diseases. Interpretation of ECG requires a bit more resource than primal vital signs. ECG is rather big data as compared to other vital signs because it is sent as a signal from the patient. ECG interpretation starts with the pinpoint operation of important waves within the signal. Those important waves are called P, Q, R, S and T waves. Heart rate information is extracted from the R waves. Consecutive R peaks are used to calculate the pulse rate. This rate is the input for the pulse interpretation. Since all waves are found as an initial step, the next operation is to determine important wave intervals. Amplitude of T wave is found by finding the mean value of the signal. Then respectively QT, QRS and PR intervals are extracted from the data [66]. At the end, those intervals and amplitudes are checked with a similar control done in primal vital sign interpretation; predefined thresholds for those amplitude and intervals determine the result of the interpretation. Figure 13 illustrates a real ECG sample with pinned waves by the system. The program is also able to extract the heart rate as it is specified in dialogbox of the figure.

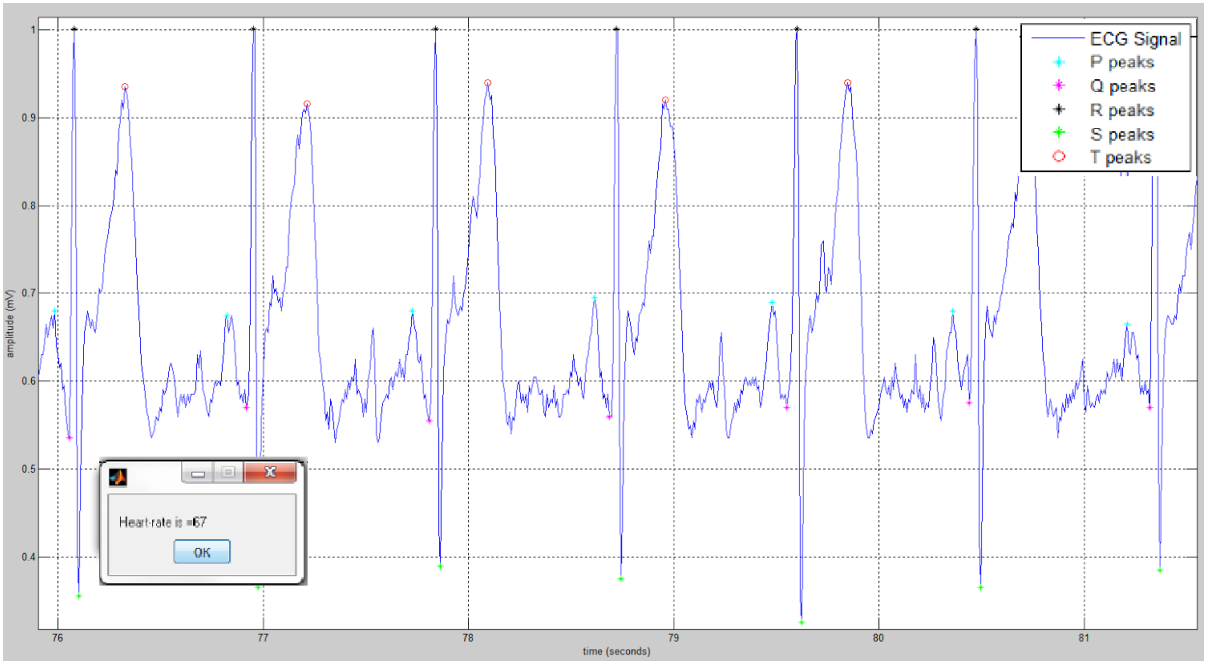


Figure 13: ECG signal and pinned waves

Consequently, under the light of the interpretation results, medical data is used to

identify whether the patient is experiencing a medical anomaly which requires immediate action to be taken or not. The detailed explanation for the identification medical anomaly is given in Section 4.5.

4.5 Critical State

Once the interpretations explained in Section 4.4 are completed, the results are examined automatically to determine potential emergency conditions. Such conditions are predefined in our model to represent a proof-of-concept. Of course, such an automatic mechanism may not give precise and accurate information about the disease. However, such a mechanism results in automated selection of the corresponding medical experts to be evoked for further examinations, tests and treatment. This is our aim in our system.

Each critical disease defined in our MAR-BAC model has certain combination of disease characteristics. Diseases defined in MAR-BAC are given in Table 2. In this table, disease name, disease characteristics and corresponding disease category are listed. If a medical data yields in any combination of disease characteristics, system automatically changes the patient's condition from *normal* to *emergency*. For example, consider the following scenario in which patient $p_i \in P$ is suffering from *Hypokalemia* at time $t \in T$. Medical data is shown as $h_{i,t} \in H$. As mentioned in Proposition 9, function $f(hi_t)$ results in set of diseases at the end of medical data interpretation. Thus, definition of the disease is $f(hi_t) = D_{i,t}$. Then disease *Hypokalemia* is going to be in this resulting set ($d_h \in D_{i,t}$). This disease belongs to the "Internal Medicine" disease category. Therefore, the function $\gamma(d_h) = c_a$ gives the corresponding disease category. Based on the category of disease, online medical experts who are specialised with the category c_a are found as the next step. If medical expert m_j is specialised in category c_a, then the function η finds m_j as a specialised medical expert in disease category. This operation is defined in Proposition 12 as if $\eta(m_j) = \gamma(d_h)$ then m_j is assumed to be more knowledgeable to cure critical disease d_h.

In the scenario exemplified above, medical experts with internal medicine specialization are selected. The selection also takes into account whether those experts are

online and authenticated or not. A given number of specialised and online medical experts are selected as medical experts to be alarmed. The number of medical experts to be notified is a system parameter. Chosen experts are notified by the system.

Algorithm 2 outlines the process of analysis and system response to criticality. In our proof of concept implementation, medical experts are notified via desktop computers; but in real life implementations this notification can be done through mobile devices. Additionally, the selected medical experts are able to monitor the changes of the patient until the patient recovers from its current emergency condition. After the patient returns to his normal condition, extra privileges given to those medical experts are revoked by the system. The privilege management according to the current condition of the patient shows the proactivity of our MAR-BAC model.

Algorithm 2 Critical State Response

Require: n: number of medical experts to be notified, $p_i \in P$, $t \in T$

```
h_{i,t} = θ(p_i, t)
D_{i_t} = f(h_{i,t})
for each d_k ∈ D_{i_t} do
    c = γ(d_k)
    doctorsToBeAlarmed = FindOnlineDoctorsWithSpeciality(c, n)
    for each m_j ∈ doctorsToBeAlarmed do
        NotifyDoctor(m_j, h_{i,t})
        if p_i ∉ P_j then
            GrantExtraPrivilege(m_j, p_i)
        end if
    end for
end for
```

Table 2: List of critical diseases

Disease Name	**Disease Characteristics**	**Disease Category**
Acidosis	• Rapid respirations • High pulse • Low blood pressure	Internal Medicine
Cardiac Tamponade	• Rapid respirations • Low pulse • Low blood pressure	Cardiology
Coronary Thrombosis	• Low respirations • Low oxygen saturation • High blood pressure	Cardiology
Hypercalcemia	• High blood pressure • ECG - Shortened QT interval	Internal Medicine
Hyperkalemia	• Low pulse • ECG - Tall T wave - Shortened QT interval - Wide QRS interval - Prolonged PR interval	Internal Medicine
Hypoglycemia	• Rapid respirations • Rapid pulse • Low blood pressure	Internal Medicine
Hypotherma	• Low respirations • Low pulse • Low blood pressure	Internal Medicine
Hypoxia	• Rapid respirations • Rapid pulse • High blood pressure • Low oxygen saturations	Pulmonology
Hypokalemia	• Low respirations • High blood pressure • Dysrythmias • ECG - Low amplitude T wave - Prolonged QT interval	Internal Medicine
Pulmonary Embolism	• Rapid respirations • Rapid pulse • Low oxygen saturation • Low blood pressure	Pulmonology
Tension Pneumothorax	• Low respirations • Low oxygen saturation • Low blood pressure	Pulmonology

5 Performance Evaluation

We have implemented our proposed MAR-BAC model using C# programming language in order to carry out a simulation-based performance evaluation. Some of the health information, such as body temperature, respiration, oxygen saturation and blood pressure have been randomly generated and used. These are also known as primal vital signs of fatal diseases. We obtained real Electrocardiogram (ECG) data for 50 different patients from the publicly available PhysioBank MIMIC II Waveform database [67]. ECG signals are interpreted using MATLAB. The interpretation of ECG also extracts the pulse rate of heart which is another vital sign. We have simulated the hospital environment in a local computer, which has Windows 7 64-bit OS, i7-2600 CPU with 3.40 GHz frequency and 8 GB RAM. User environment is simulated with the help of another computer which has Windows 7 64-bit OS, i5-3230M CPU with 2.60 GHz frequency and 4GB RAM.

5.1 Performance Metrics and Parameters

The performance analysis of our MAR-BAC model consists of two main parts; server processing delay and end-to-end delay. Server processing delay is the delay due to the operations performed at the server side and consists of three parts (i) ECG interpretation delay, (ii) database operations delay and (iii) security and access control delay. The ECG interpretation part is the one explained in Section 4.4. Interpretation of ECG signal is done by first finding QRS complex intervals from the signal. Afterwards, interval changes and other medical data are statically analysed and interpreted by the system. Database operations are insert/update and query operations. Security delay is the time to perform security operations on the medical data. The security opera-

tions are symmetric encryption/decryption for confidentiality and computation of hash based message authentication code (HMAC) for the message integrity. Access control delay is the time to perform access related processing and server operations which are used to maintain the server. In case of medical analysis, server preparation of medical data and after analysis, server actions to respond clients accordingly are also covered in access control delay. End-to-end delay is total time needed for the entire processing, data transmission and receiving acknowledgements. These time related metrics are important since in a healthcare system, we need fast responses in order to manage critical states. Moreover, the bottleneck of analysis is the interpretation of ECG signal. Thus we also measure it separately.

System parameters defined in our system are: (i) number of on-line patients, (ii) medical data analysis request interval per user and (iii) physical location of clients. All three parameters determine the system load. Medical data analysis request interval per user is modelled using *exponential distribution* [68] with average request rate parametrised as λ. Exponential distribution is used since the ECG analyses create a queue at the server side and in the queuing systems interarrival times are generally modelled with exponential distribution. Since almost the entire duration of medical data interpretation is utilised for ECG analysis, in the rest of this thesis, we will refer to medical data interpretation and ECG analysis/interpretation interchangeably for performance point of view. The raise in number of patients also increases the amount of medical information sent to the hospital server. Access requests to medical data, which are sent by medical experts, also generate system load. Whenever a patient sends its medical data to the system, the record is queued by the system for medical analysis. Medical analysis subsystem dequeues medical data one by one from the head of the queue and analyses each them. The physical location of the clients is categorised as *local* and *remote*. A local client is the one who connects to the hospital server from the same network. A remote client is connected to hospital server from a different network.

The most memory consuming medical data is ECG. Although all ECG data is taken as 5 minute measurement, there are variations in data length of ECG signals. A set of ECG signal may take as much as 1 MByte. As it is mentioned before, ECG requires a

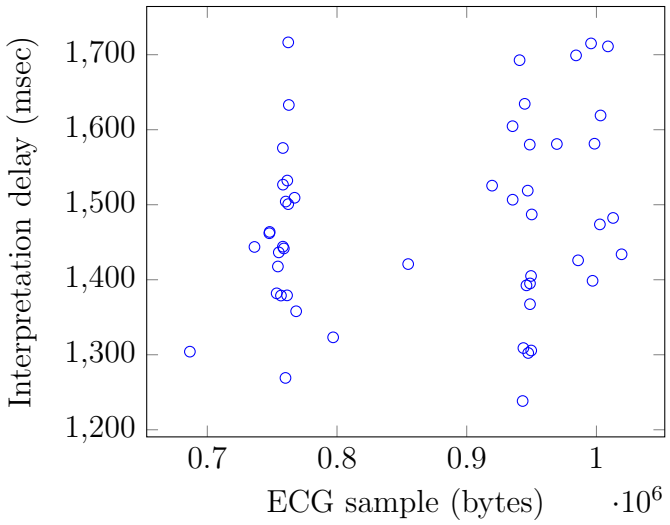


Figure 14: ECG interpretation Delay

special interpretation and processing. Figure 14 shows time variations of ECG signal processing and interpretation among 50 patients. As can be seen in this figure, ECG signal interpretation delay varies between 1200 and 1700 msec and is independent of the length of the sample. Thus, it changes from patient to patient due to the characteristics within the ECG signal.

5.2 Simulation Results

Simulation results of our MAR-BAC system are explained in this subsection. In all of the analyses, we scaled the number of patients from 100 to 500. Each ECG data has its own characteristics which changes time required to complete interpretation. Thus, there is a jitter here. In order to model the queue of ECG interpretation at the server side, we use various average ECG signal interpretation request rate (λ) values. Finally, we performed tests for patients in local and remote networks.

5.2.1 Analysis of Secure Login protocol

This subsection gives the timings of login protocol, which was explained in Section 4.2. The analysis is performed for local and remote users. In the Table 3, the results are given. Time to perform mutual authentication, ticket generation and ticket

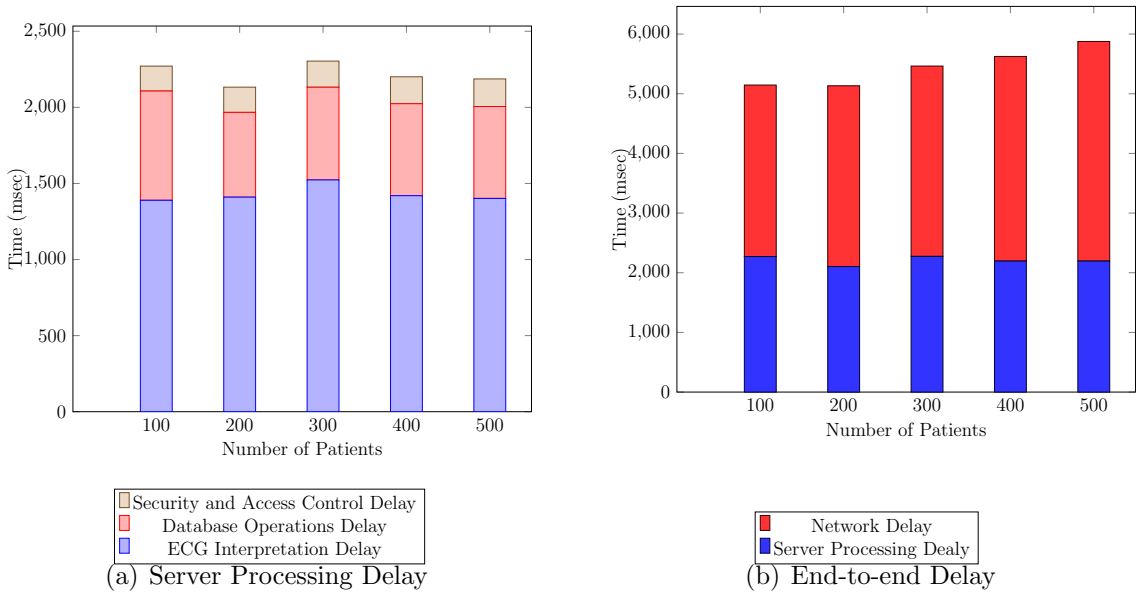
validation is 100 ms per user in a local area network and 214 ms per user connected remotely. Since this is performed only once per session, this delay is not so significant in overall performance.

Table 3: Simulation result for login protocol proposed in Section 4.2

	Local User	Remote User
Authentication, login and ticket validation delay	100 ms	217 ms

5.2.2 Scalability analysis of local patients

In this subsection, the simulation is performed in a local area network. In the first set of tests, we set the λ value to 0.0001 $requests/(sec * user)$. Figure 15 gives the simulation results. As shown in the figure, server side operations are between 2 and 2.5 seconds and end-to-end delay is between 5 and 6 seconds. The increase in number of patients does not significantly affect the timings. The reason behind this behaviour is low request rate and consequently no waiting time in the queues for medical analysis.



(a) Server Processing Delay (b) End-to-end Delay

Figure 15: Simulation results with local patients, $\lambda = 0.0001$ $requests/(sec * user)$

The performance results of Figure 15 are also shown in Table 4 in tabular format. Moreover, the percentages of delay components are also shown in this table. As ob-

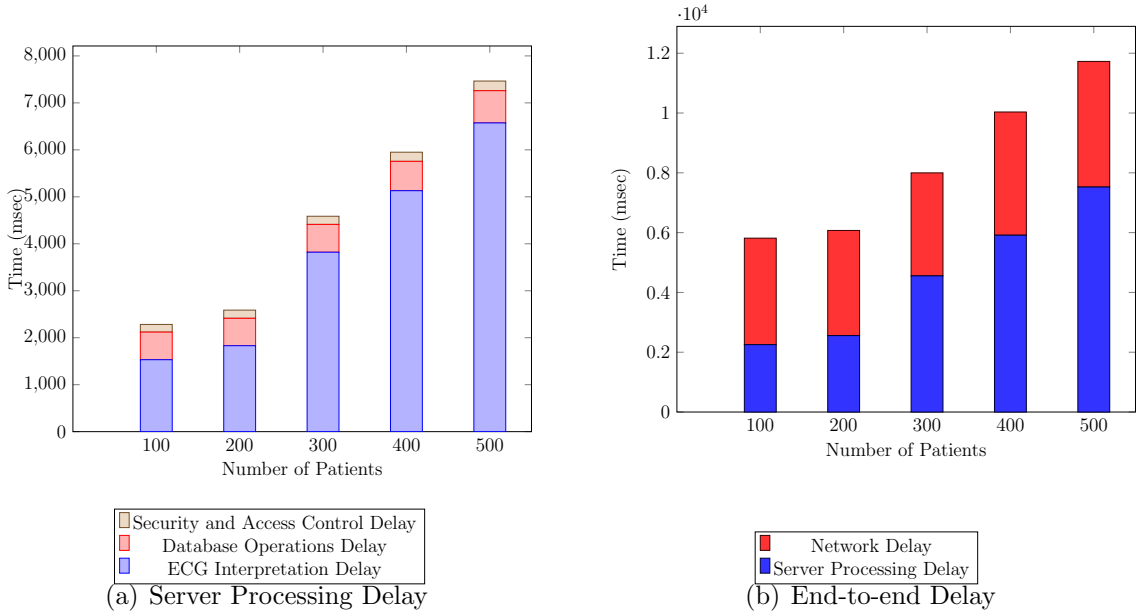
served from this table, more than half of the total delay is due to the network delay. Interpretation and analysis of medical data nearly takes quarter of the whole delay. Security related and access control operations take less than 4% of the total delay. Thus we conclude that security and access control bring only a negligible overhead in this setting. Moreover, interpretation is not dependent on server load because request rate is low in this simulation setup.

Table 4: Simulation result with local patient execution timing and percentages; $\lambda = 0.0001\ requests/(sec * user)$.

	Number of Patients									
	100		200		300		400		500	
	ms	%	ms	%	ms	%	ms	%	ms	%
Interpretation delay	1389.4	27.00	1410.0	27.31	1523.1	27.72	1419.5	25.23	1401.7	23.89
Database operations delay	717.9	13.96	557.1	10.79	609.2	11.09	599.4	10.66	603.8	10.29
Security and Access Control delay	163.6	3.17	165.6	3.20	171.6	3.13	176.8	3.14	181.2	3.08
Network delay	2874.6	55.87	3029.9	58.68	3189.7	58.06	3429.1	60.97	3680.2	62.72
Total delay	5145.5	100	5162.6	100	5493.6	100	5624.8	100	5866.9	100

Second set of tests is performed with a moderate ECG request rate, $\lambda = 0.001$ $requests/(sec * user)$. Figure 16 shows the increase in interpretation time of medical data with increased number of patients. Server side delay is measured between 2 and 8 seconds and end-to-end delay is between 6 and 12 seconds. The reason of this increase is the queuing effect. The load of the system creates to a queue of access requests at the hospital server (ADPS) and the waiting time in this queue causes the increased delay.

The performance results of Figure 16 are also shown in Table 5 in tabular format. As compared to the previous setting, time for interpretation has increased from quarter of the total delay to more than half of it for larger number of patients. The time required to perform database operations and security and access control delay do not change significantly with different number of patients. In this setting, the share of security and access control delay is less 3% of the total delay. Another increased delay component of this setting is the network delay. This is due to the fact that increase in λ causes more medical data to be transmitted and more network level interrupts at the server



(a) Server Processing Delay (b) End-to-end Delay

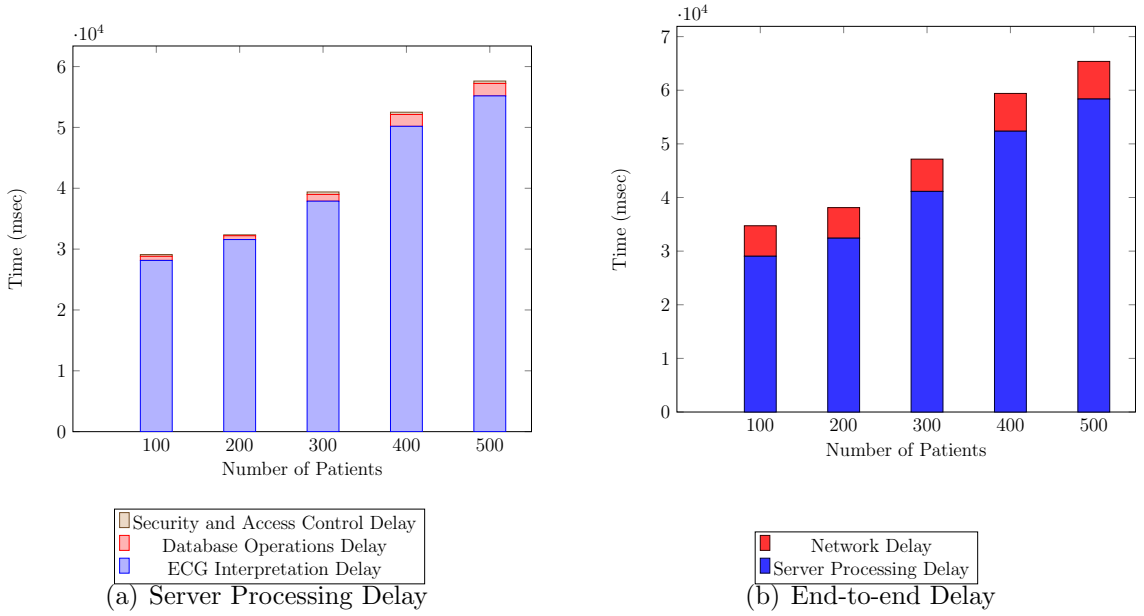
Figure 16: Simulation results with local patients, $\lambda = 0.001$ $requests/(sec * user)$

Table 5: Simulation result with local patient execution timing and percentages; $\lambda = 0.001$ $requests/(sec * user)$.

	Number of Patients									
	100		200		300		400		500	
	ms	%	ms	%	ms	%	ms	%	ms	%
Interpretation delay	1531.8	26.21	1829.7	29.97	3820.7	47.58	5130.2	50.94	6573.1	56.35
Database operations delay	587.7	10.05	583.9	9.57	591.6	7.37	624.9	6.20	686.0	5.88
Security and Access Control delay	163.3	2.79	174.1	2.85	173.7	2.16	194.7	1.95	205.7	1.76
Network delay	3562.0	60.95	3517.0	57.61	3444.6	42.89	4120.5	40.91	4200.9	36.01
Total delay	5844.8	100	6104.7	100	8030.6	100	10070.3	100	11665.7	100

side. Especially for large number of patients (400 and 500 patients), the network delay significantly increases. However, as the interpretation delay increases faster for these numbers of patients, the share of network delay in total delay decreases.

In final set of simulations with local patients, we set λ to 0.01 $request/(secs * user)$. The results of the simulation can be seen in Figure 17. Server side delay is measured between 30 and 58 seconds and end-to-end delay is between 35 and 65 seconds. The main reason of these large delay values is the increased system load due to larger λ



(a) Server Processing Delay (b) End-to-end Delay

Figure 17: Simulation results local patients, $\lambda = 0.01\ requests/(sec * user)$

value. This system load especially increases interpretation and network delays.

The performance results of Figure 17 are also shown in Table 6 in tabular format. Interpretation and medical analysis dominates the most of total delay. Other delay components also increase in parallel with high system load. However, as the interpretation delay increase is much more significant, the shares of other delay components get smaller in this simulation setting as compared to other ones. It is remarkable that the share of security and access control delay is less than 1%.

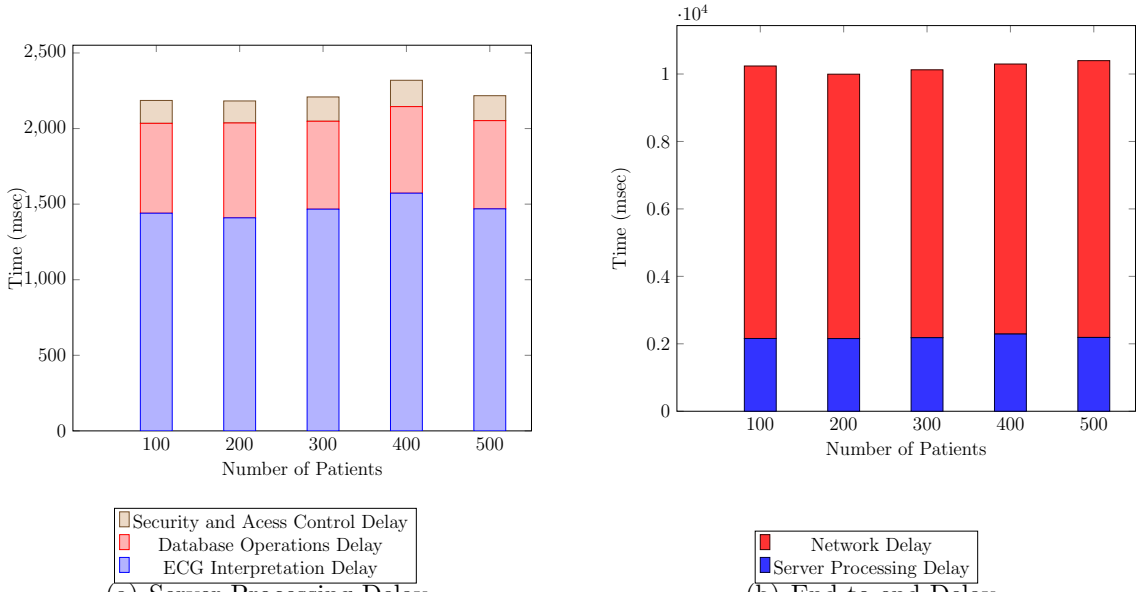
Table 6: Simulation result with local patient execution timing and percentages; $\lambda = 0.01\ requests/(sec * user)$.

	Number of Patients									
	100		200		300		400		500	
	ms	%	ms	%	ms	%	ms	%	ms	%
Interpretation delay	28142.7	80.89	31417.3	82.43	39719.4	84.23	50181.5	84.28	55181.5	85.38
Database operations delay	675.2	1.94	705.8	1.85	1099.1	2.33	1948.3	3.27	2048.3	3.17
Security and Access Control delay	288.7	0.83	291.4	0.76	317.7	0.67	369.5	0.62	390.5	0.60
Network delay	5685.9	16.34	5701.9	14.96	6017.6	12.77	7044.0	11.83	7014.0	10.85
Total delay	34792.5	100	38116.4	100	47153.8	100	59543.3	100	64634.3	100

5.2.3 Scalability analysis of remote patients

In this subsection, the simulation is performed with patients in a non-local (remote) network. First set of tests for remote patients are performed with $\lambda = 0.0001$ $request/(sec * user)$. Figure 18 shows the results of this simulation. The delay in server side is between 2 and 2.5 seconds and end-to-end delay is between 10 and 10.5 seconds. As in the local user tests with the same λ value, the system load is very low in this setting. Thus we do not observe any queuing effect as the number of patients increase. Therefore, the delay figures do not change significantly with increased number of patients.

The most significant outcome of the remote patient setting is that network delay increases significantly. Here, the bottleneck is the client side due to slower connection as compared to the server side.



(a) Server Processing Delay (b) End-to-end Delay

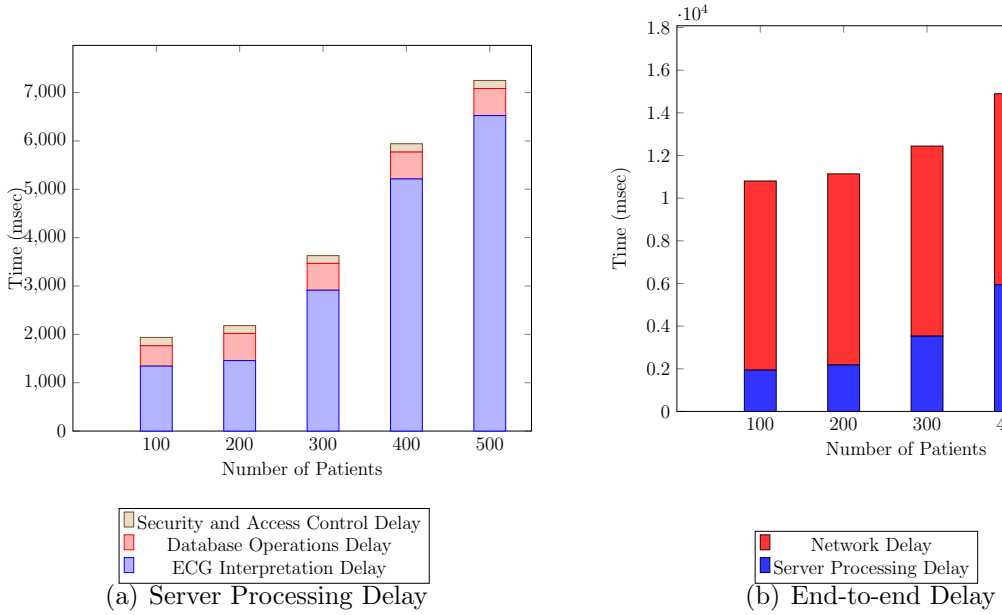
Figure 18: Simulation results remote patients, $\lambda = 0.0001$ $requests/(sec * user)$

The performance results of Figure 18 are also shown in Table 7 in tabular format. Network delay overwhelmingly dominates the rest of delay components of the total delay ($\approx$ 80% of the total delay). The database operations delay is around 5% of the total delay. The delay caused by MAR-BAC security and access control is less than 2%, which is quite insignificant.

Table 7: Simulation result with remote patient execution timing and percentages; $\lambda = 0.0001\ requests/(sec * user)$.

	Number of Patients									
	100		200		300		400		500	
	ms	%	ms	%	ms	%	ms	%	ms	%
Interpretation delay	1439.9	14.02	1409.2	14.06	1466.7	14.45	1572.4	15.23	1468.7	14.09
Database operations delay	594.5	5.79	627.6	6.26	581.5	5.73	572.7	5.55	583.7	5.60
Security and Access Control delay	151.3	1.48	145.6	1.46	160.7	1.58	174.1	1.68	164.9	1.58
Network delay	8080.1	78.71	7839.5	78.22	7943.1	78.24	8006.6	77.54	8207.6	78.73
Total delay	10265.8	100	10022.4	100	10152.1	100	10325.8	100	10424.9	100

Second set of tests for remote clients is performed with moderate medical data interpretation request rate, $\lambda = 0.001\ request/(secs * user)$. Figure 19 shows the results of this simulation. The server processing delay is between 2 and 7 seconds and end-to-end delay is between 10 and 16.5 seconds. As the number of patients increase, both delay values increase. The reason of this behaviour is the queuing effect with increased system load at this λ value, as in the local user case.



(a) Server Processing Delay (b) End-to-end Delay

Figure 19: Simulation with remote patients given $\lambda = 0.001\ requests/(sec * user)$

The test results given in Figure 19 are detailed in Table 8. As can be seen there,

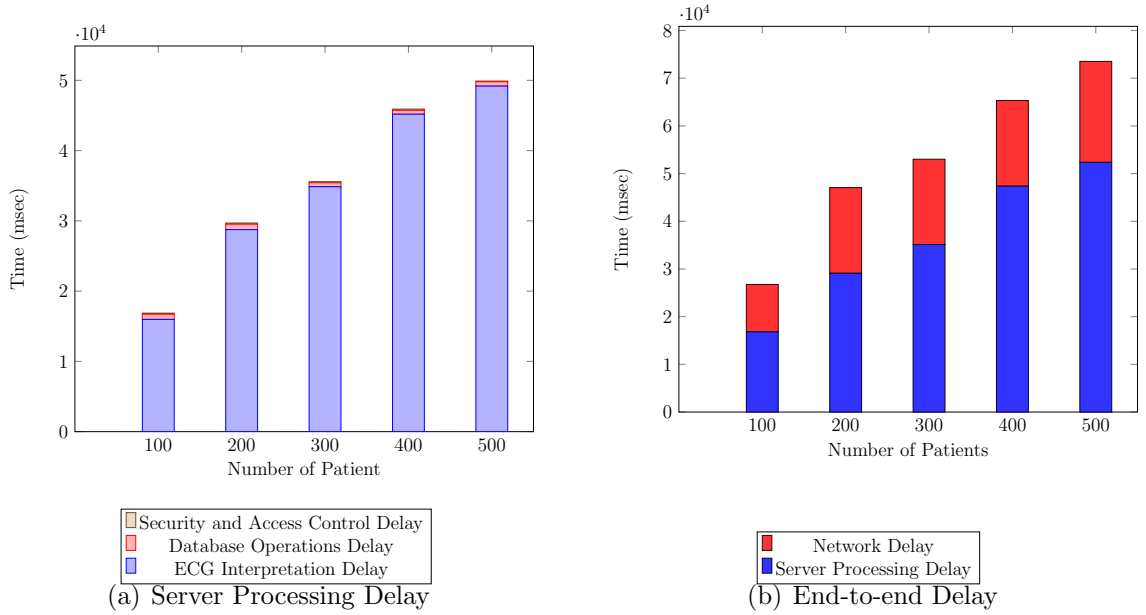
network delay is the largest component of the total delay, although its percentage reduces as the number of patients increase. This is due to increased ECG interpretation delay with increased number of patients. Although system load increases in parallel with the number of patients, the time required to perform security and access operations has not been affected too much. This is because security overhead mainly relies on the size of the data and access control delay is not affected by the size of data transmitted.

Table 8: Simulation result with remote patient execution timing and percentages; $\lambda = 0.001\ requests/(sec * user)$.

	Number of Patients									
	100		200		300		400		500	
	ms	%	ms	%	ms	%	ms	%	ms	%
Interpretation delay	1343.0	12.43	1457.8	13.09	2912.8	23.22	5213.8	35.00	6524.7	39.69
Database operations delay	421.2	3.90	560.6	5.03	554.9	4.42	555.9	3.73	559.6	3.40
Security and Access Control delay	172.5	1.6	162.2	1.46	160.3	1.28	172.4	1.16	166.3	1.02
Network delay	8867.7	82.07	8959.6	80.42	8912.7	71.08	8955.7	60.11	9188.6	55.89
Total delay	10804.4	100	11140.2	100	12540	100	14897.8	100	16439.2	100

The final set of simulations for remotely connected patients is performed with the $\lambda = 0.01\ request/(secs * user)$. Figure 20 shows the results of this simulation. At the server side, delay is between 17 and 50 seconds. The end-to-end delay is between 27 and 71 seconds. Since the system is heavily loaded with the λ value, ECG interpretation and consequently server processing delay increase dramatically as compared to other settings. Moreover, in this setting, server side processing delay becomes larger than the network delay. Other than that, time required to analyse medical data increases as number of clients increases due to the same queuing effect.

The test results of Figure 20 are also given in Table 9 in tabular format. The interpretation delay started with nearly with 60% of the total delay and increased up to around 70% as the number of patients grows. Security and access control delay is less than 1% of the total delay, which is insignificant. Moreover, network delay increases as the number of patients increase.



(a) Server Processing Delay (b) End-to-end Delay

Figure 20: Simulation with remote patients given $\lambda = 0.01\ requests/(sec * user)$

Table 9: Simulation result with remote patient timing and execution percentages; $\lambda = 0.01\ requests/(sec * user)$.

	Number of Patients									
	100		200		300		400		500	
	ms	%	ms	%	ms	%	ms	%	ms	%
Interpretation delay	15979.6	59.62	28753.7	60.32	34857.1	65.17	45181.5	70.75	49181.5	69.21
Database operations delay	748.2	2.79	779.5	1.64	563.7	1.06	555.9	0.87	618.8	0.87
Security and Access Control delay	142.5	0.53	162.2	0.31	163.9	0.30	172.4	0.26	97.6	0.14
Network delay	9934.2	37.06	17971.9	37.70	17902	33.47	17955.7	28.12	21165.1	29.78
Total delay	26804.5	100	47667.3	100	53486.7	100	63865.5	100	71063	100

5.3 Memory Requirements Analysis

In this subsection, memory requirements of our MAR-BAC model are analysed. MAR-BAC is an extension of RBAC model and it needs storage to operate on access rights. Our storage requirements for MAR-BAC are grouped under 3 main categories: (i) client, (ii) ATOS and (iii) ADPS memory requirements.

Memory requirements of one client are given in Table 10. Each client has its own unique identifier which takes about 20 bytes of memory. Nonce and session key (K_{CS})

are used for secure login protocol explained in Section 4.2. 32 bytes of storage is required for both of those values. Role of the client also consumes memory at client side which is about 20 bytes. Because of each client connects to ATOS and ADPS, 2 different sockets are needed and each socket consumes 94 KBytes of memory. Another important memory consuming entity is the RSA public key of ATOS. We choose public key as 2048-bit; thus, memory requirement of ATOS public key is about 500 bytes because of the key structure used. In total, a client requires approximately 189 Kbytes of memory.

Table 10: Memory requirement of one client

	Memory
Client id	≈20 Bytes
Client-ADPS session key (K_{CS})	32 Bytes
Client nonce	32 Bytes
Client role	≈20 Bytes
Socket for ATOS communication	94 KBytes
Socket for ADPS communication	94 KBytes
Public Key of ATOS	≈500 Bytes

ATOS unit memory requirements are given in Table 11. ATOS keeps its RSA public-private key pair which takes 2 KBytes of storage. ATOS also keeps session keys between different hospital servers (ADPS) which is 32 bytes per server. Since ATOS is responsible for authentication of clients, it keeps id, pin, OTP, nonce and session keys of each client. In total, those values requires approximately 109 bytes of memory for each client connected to ATOS. ATOS also keeps sockets for each client for communication operations, which takes 94 KBytes of memory.

To sum up, Equation 1 gives the total memory required by ATOS.

$$\begin{aligned} M_{ATOS} = 2 + ((n_{adps} \times 32) \times 10^{-3}) \\ + ((n_c \times 109) \times 10^{-3}) + (n_c \times 94) \ \ KBytes \end{aligned} \tag{1}$$

where, n_{adps} is the number of ADPS servers connected to ATOS and n_c is the number of clients who are connected to ATOS.

Table 11: Memory Requirement of ATOS (unit values)

	Memory
RSA public-private key pair	2 KBytes
ATOS-ADPS session key (K_{AA})	32 Bytes per ADPS server
Client id	≈20 Bytes per client
Client pin	≈10 Bytes per client
Client OTP	≈15 Bytes per client
Client nonce	32 Bytes per client
Client-ADPS session key (K_{CS})	32 Bytes per client
Client Socket	94 KBytes per client

ADPS unit memory requirements are given in Table 12. ADPS securely communicates with ATOS using a session key (K_{AA}), which is 32 bytes. There is also a socket for this communication at ADPS side which uses 94 KBytes of memory. For each client, ADPS needs to keep track of client id, session key (K_{CS}), role, and socket information which require about 95 KBytes of memory. As it is stated in Section 4.3, we designed the Access Policy Manager (APM) for regulating access control operations. In APM, role-participation mappings and activity cell mappings, which are basically participation-act mappings, require 40 bytes for each mapping. Because of MAR-BAC is capable of interpreting and analysing the medical data, ADPS keeps a disease list and a symptom-disease mapping whether a patient is suffering from a medical emergency. 40 bytes needed for keeping each disease information and each symptom-disease mapping requires around 2 KBytes of memory.

Total memory requirement of ADPS is calculated in Equation 2.

$$\begin{aligned} M_{ADPS} = {} & (32 \times 10^{-3}) + 94 \\ & + ((n_c \times 102) \times 10^{-3}) + (n_c \times 94) \\ & + ((n_r \times 30) \times 10^{-3}) \ + ((n_{r,p} \times 40) \times 10^{-3}) + ((n_{ac} \times 40) \times 10^{-3}) \\ & + ((n_d \times 40) \times 10^{-3}) + (n_d \times 2) \ KBytes \end{aligned} \tag{2}$$

where, n_r is the number of roles, $n_{r,p}$ is the number of role-participation mappings, n_{ac} is the number of activity cells, n_d is the number of diseases and n_c is the number of clients connected to ADPS.

Table 12: Memory Requirement of ADPS (unit values)

	Memory
ATOS session key (K_{AA})	32 Bytes
Socket for ATOS communication	94 KBytes
Client id	≈20 Bytes per client
Client-ADPS session key (K_{CS})	32 Bytes per client
Client role	≈20 Bytes per client
Client Socket	94 KBytes per client
Role List	≈30 Bytes per role
Role-Participation mapping	40 Bytes per mapping
Activity Cell	40 Bytes per mapping
Disease List	40 Bytes per disease
Symptom-Disease mapping	2 KBytes per mapping

In our simulations, we used MATLAB for ECG interpretation. The MATLAB object, which is defined at ADPS side, consumes 154 MBytes of memory in order to operate ECG interpretation. We have also used ECG samples which approximately consume 1MByte of memory per sample. These samples use memory at both client side and ADPS side because of the assumption of client is generating the medical data and sends it to ADPS.

5.4 Comparative Analysis with the Related Work

In this part, some related work in the literature are explained and compared with MAR-BAC model.

Muppavarapu et al. [1] aim to reduce the administration overhead by dividing work to different sections of the access control model. To achieve this division of operations, they designed an access control model which retrieves capabilities of a user from an entity called Shibboleth. In order to operate over certain resources, user should get its capabilities from Shibboleth and asks for access permission to a resource manager. In Figure 21, architecture of access control model of Shibboleth is given. In [1], all user-role assignments are controlled by Shibboleth which is considered to be secure. However, it is not clearly shown that the security of the messages is maintained during transmission of access request and access response. In other words, confidentiality or

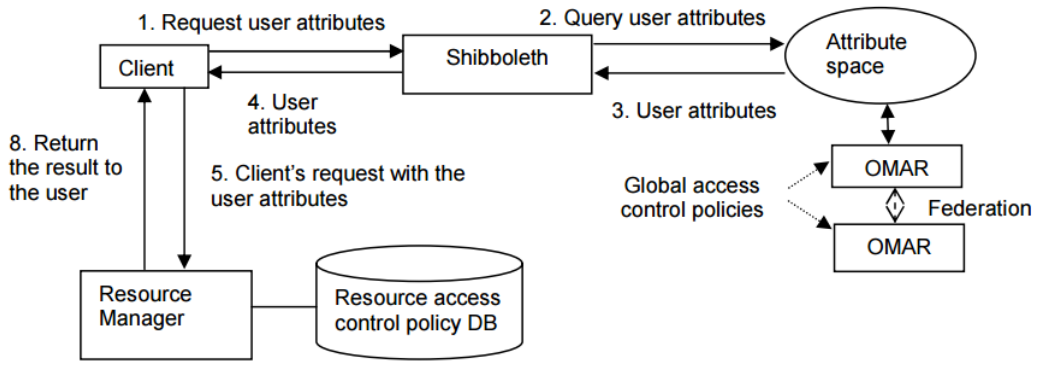


Figure 21: Shibboleth architecture in work [1]

integrity of the messages are not well defined in this system. Even the communication is safeguarded with a secure channel, security of the access is not clearly defined. We overcome those problems in MAR-BAC model since confidentiality and integrity of the messages transmitted are safeguarded with the establishment of secure channel as explained in Section 4.2. Another drawback for [1] is the requirement of receiving a ticket-like access permission for each distinct request. They claimed that the clients are able to receive a single ticket which permits them to operate all valid operations within the system. This is the same as receiving capabilities which was described in Section 2.1.1. However, receiving capabilities, in order to access certain objects, is vulnerable to forgery which causes problems such as modification of capabilities by the user. If those capabilities of a user can be modified, then user can access resources which are not open for that particular user. The problem of revocation of clients after receiving access capabilities from Shibboleth and problem of cloning those capabilities are also not addressed in [1]. In MAR-BAC, access permissions are directly controlled by APM. Therefore users are able to request access for certain operations. APM contains the information about access rights, which means predefined operations which can be permitted to users in APM. As a result, the problems of cloning and revocation of capabilities do not interfere with regular access control operations regulated by MAR-BAC.

The security overhead reported in [1] is limited to a single command and is 270 ms. Even this incomplete timing value is greater than our MAR-BAC's worst case secure

login protocol execution time, which is 220 ms. Thus, we conclude that our MAR-BAC outperforms the work proposed in [1].

Another inspiring study is done by Venkatasubramanian [2]. This work utilises information gathering and transmission of it by applying security mechanisms in order to protect the data. The data taken from patients are sent to a hospital server. In the server, there exists a continuous control mechanism which scans for the patient's health information in certain periods of time. If the system finds an anomaly about a patient, it automatically takes a responsive action. Table 13 gives the timing results which are taken from the implementation of [2]. Physiological signals and their timing results of performing some computations and corresponding current information are included in this table. In order to apply a secure model, patients' physiological signal features are extracted and used for secret key computations. Those features, called as chaff points, assist to apply fuzzy vault [69] based cryptography. However, generation of such keys requires more than 15 seconds at both sender and receiver sides. In our model, security overhead is always less then 0.5 second. This means the work in [2] is at least 30 times slower than our MAR-BAC model from security overhead point of view.

Table 13: Timing results from Venkatasubramanian work [2]

Mote	Stage	Current Draw (Radio-Off)	Current Draw (Radio-On)	Time (msec)
Sender/Receiver	Sensing	6.6mA	6.6mA	12700
	FFT Computation	1mA	19.56mA	2138
	Peak Detection and Quantiz.	0.14mA	18.72mA	12.4
	Feature Generation	0.11mA	18.72mA	13.6
Sender	Polynomial Gen. & Eval.	0.08mA	18.68mA	8
	Chaff Points Gen.	0.01mA	18.61mA	14 per 10 points
	Vault Tx	-	19.33mA	1350(1K), 2700(2K), 4000(3K), 5360(4K), 6750(5K)
	Ack Rx	-	19.20mA	20
Receiver	Vault Rx	-	19.41mA	1400(1K), 2750(2K), 4100(3K), 5370(4K), 6760(5K)
	Lagrangian Interpolation	0.43mA	19.04mA	50
	Ack Tx	-	19.11mA	17

6 Conclusions

In this thesis, we designed a Medically Adaptive Role Based Access Control (MAR-BAC) model for healthcare systems. In our model, we extended classical RBAC by adding dynamicity with proactively change of access definitions. While doing such modifications, care has been taken not to violate the privacy of the content defined in the system. Our work is capable to transmit medical information over a public channel. We have designed a protocol in order to establish a secure channel. Another plus for our system is the ability of interpreting and analysing the medical data. In this way, the system finds out emergency conditions of the patients. Medical anomalies, which are interpreted as dangerous for the patient health, are notified to doctors in an automated way. It has been defined by which access control policies are applied in the system. Additionally, users are able to request actions which are valid in different roles. This increases the dynamicity of the system. By applying such a mechanism, private information disclosure and unauthorised accesses are avoided.

We implemented MAR-BAC model and performed simulation based performance analysis. The medical data generated from a single patient does not actually change immensely before 45 minutes. However, in our simulations we used smaller medical data generation intervals to test the limits of MAR-BAC. Even in such a case, end-to-end delay for a patient to be responded in case of an emergency is at most around a minute and scales linearly with respect to the number of patients. Most of this time is spent for medical data interpretation and transmission; the overhead of security is less than 1%, which is not so significant. We also compared our MAR-BAC model's security overhead with related work and showed that our security overhead is much smaller.

References

[1] V. Muppavarapu and S. M. Chung, "Role-based access control for cyber-physical systems using shibboleth," in *Proceedings of DHS Workshop on Future Directions in Cyber-Physical Systems Security*, 2009, pp. 57–60.

[2] K. K. Venkatasubramanian, "Security solutions for cyber-physical systems," Ph.D. dissertation, Arizona State University, 2009.

[3] P. Samarati and S. C. de Vimercati, "Access control: Policies, models, and mechanisms," in *Foundations of Security Analysis and Design*. Springer, 2001, pp. 137–196.

[4] F. M. Kugblenu and M. Asim, "Separation of duty in role based access control system: A case study," Ph.D. dissertation, MS Thesis, Thesis no: MCS-2006: 16, 2007.

[5] D. Ferraiolo, D. R. Kuhn, and R. Chandramouli, *Role-based access control*. Artech House, 2003.

[6] R. Slade, *Dictionary of Information Security*. Syngress, 2006.

[7] R. S. Sandhu and P. Samarati, "Access control: principle and practice," *Communications Magazine, IEEE*, vol. 32, no. 9, pp. 40–48, 1994.

[8] M. A. Harrison, W. L. Ruzzo, and J. D. Ullman, "Protection in operating systems," *Communications of the ACM*, vol. 19, no. 8, pp. 461–471, 1976.

[9] C. S. Jordan, *Guide to Understanding Discretionary Access Control in Trusted Systems*. DIANE Publishing, 1987.

[10] H. Lindqvist, "Mandatory access control," *Master's Thesis in Computing Science, Umea University, Department of Computing Science, SE-901*, vol. 87, 2006.

[11] B. W. Lampson, "Protection," *ACM SIGOPS Operating Systems Review*, vol. 8, no. 1, pp. 18–24, 1974.

[12] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein *et al.*, *Introduction to algorithms*. MIT press Cambridge, 2001, vol. 2.

[13] V. C. Hu, D. R. Kuhn, T. Xie, and J. Hwang, "Model checking for verification of mandatory access control models and properties," *International Journal of Software Engineering and Knowledge Engineering*, vol. 21, no. 01, pp. 103–127, 2011.

[14] M. Nyanchama and S. L. Osborn, "Modeling mandatory access control in role-based security systems." in *DBSec*. Citeseer, 1995, pp. 129–144.

[15] I. Ray and M. Kumar, "Towards a location-based mandatory access control model," *Computers & Security*, vol. 25, no. 1, pp. 36–44, 2006.

[16] E. M. Geepalla, "Model-driven approaches to analysing time-and location-dependent access control specifications," Ph.D. dissertation, University of Birmingham, 2013.

[17] D. E. Bell and L. J. LaPadula, "Secure computer systems: Mathematical foundations," DTIC Document, Tech. Rep., 1973.

[18] K. J. Biba, "Integrity considerations for secure computer systems," DTIC Document, Tech. Rep., 1977.

[19] R. S. Sandhu, "Lattice-based access control models," *Computer*, vol. 26, no. 11, pp. 9–19, 1993.

[20] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role-based access control models," *Computer*, vol. 29, no. 2, pp. 38–47, 1996.

[21] S. Chakraborty and I. Ray, "Trustbac: integrating trust relationships into the rbac model for access control in open systems," in *Proceedings of the eleventh ACM symposium on Access control models and technologies*. ACM, 2006, pp. 49–58.

[22] D. F. Ferraiolo, J. F. Barkley, and D. R. Kuhn, "A role-based access control model and reference implementation within a corporate intranet," *ACM Transactions on Information and System Security (TISSEC)*, vol. 2, no. 1, pp. 34–64, 1999.

[23] M. D. M. Gilbert, "An examination of federal and commercial access control policy needs," in *National Computer Security Conference, 1993 (16th) Proceedings: Information Systems Security: User Choices*. DIANE Publishing, 1995, p. 107.

[24] T. Mayfield, J. E. Roskos, S. R. Welke, J. M. Boone, and C. W. McDonald, "Integrity in automated information systems," DTIC Document, Tech. Rep., 1991.

[25] E. Bertino, B. Catania, M. L. Damiani, and P. Perlasca, "Geo-rbac: a spatially aware rbac," in *Proceedings of the tenth ACM symposium on Access control models and technologies*. ACM, 2005, pp. 29–37.

[26] Y. Xuexiong, W. Qinxian, and X. Changzheng, "A multiple hierarchies rbac model," in *Communications and Mobile Computing (CMC), 2010 International Conference on*, vol. 1. IEEE, 2010, pp. 56–60.

[27] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, "Towards a better understanding of context and context-awareness," in *Handheld and ubiquitous computing*. Springer, 1999, pp. 304–307.

[28] B. Schilit, N. Adams, and R. Want, "Context-aware computing applications," in *Mobile Computing Systems and Applications, 1994. WMCSA 1994. First Workshop on*. IEEE, 1994, pp. 85–90.

[29] D. Kulkarni and A. Tripathi, "Context-aware role-based access control in pervasive computing systems," in *Proceedings of the 13th ACM symposium on Access control models and technologies*. ACM, 2008, pp. 113–122.

[30] R. Bhatti, E. Bertino, and A. Ghafoor, "A trust-based context-aware access control model for web-services," *Distributed and Parallel Databases*, vol. 18, no. 1, pp. 83–105, 2005.

[31] M. J. Covington, W. Long, S. Srinivasan, A. K. Dev, M. Ahamad, and G. D. Abowd, "Securing context-aware applications using environment roles," in *Proceedings of the sixth ACM symposium on Access control models and technologies*. ACM, 2001, pp. 10–20.

[32] S. K. Gupta, T. Mukherjee, and K. Venkatasubramanian, "Criticality aware access control model for pervasive applications," in *Pervasive Computing and Communications, 2006. PerCom 2006. Fourth Annual IEEE International Conference on.* IEEE, 2006, pp. 5–pp.

[33] J. D. Woodward, N. M. Orlans, and P. T. Higgins, *Biometrics:[identity assurance in the information age].* McGraw-Hill/Osborne New York, 2003.

[34] I. Altman, "Privacy: A conceptual analysis." *Environment and behavior*, vol. 8, no. 1, pp. 7–29, 1976.

[35] N. Serenko and L. Fan, "Patients perceptions of privacy and their outcomes in healthcare," *International Journal of Behavioural and Healthcare Research*, vol. 4, no. 2, pp. 101–122, 2013.

[36] J. Al-Muhtadi, R. Hill, and S. Al-Rwais, "Access control using threshold cryptography for ubiquitous computing environments," *Journal of King Saud University-Computer and Information Sciences*, vol. 23, no. 2, pp. 71–78, 2011.

[37] V. Stanford, "Pervasive health care applications face tough security challenges," *pervasive computing, IEEE*, vol. 1, no. 2, pp. 8–12, 2002.

[38] X. Jiang and J. A. Landay, "Modeling privacy control in context-aware systems," *Pervasive Computing, IEEE*, vol. 1, no. 3, pp. 59–63, 2002.

[39] H. Chan and A. Perrig, "Security and privacy in sensor networks," *computer*, vol. 36, no. 10, pp. 103–105, 2003.

[40] K. O'neil and G. R. Seidman, "Personal information security and exchange tool," Nov. 16 1999, uS Patent 5,987,440.

[41] L. Bickley and P. G. Szilagyi, *Bates' guide to physical examination and history-taking.* Lippincott Williams & Wilkins, 2012.

[42] D. M. Anderson, L. E. Anderson, and W. D. Glanze, *Mosby's medical dictionary.* Mosby St. Louis, MO, 2002.

[43] S. James, "Gale encyclopedia of medicine," *Reference Reviews*, vol. 16, no. 8, pp. 40–41, 2002.

[44] "Farlex partner medical dictionary," http://medical-dictionary.thefreedictionary.com/oxygen+saturation, 2012, accessed at 1 May 2015.

[45] J. Mooney, *Illustrated Dictionary of Podiatry and Foot Science*. Elsevier Health Sciences, 2009.

[46] R. B. Devereux, T. G. Pickering, G. A. Harshfield, H. D. Kleinert, L. Denby, L. Clark, D. Pregibon, M. Jason, B. Kleiner, J. S. Borer *et al.*, "Left ventricular hypertrophy in patients with hypertension: importance of blood pressure response to regularly recurring stress." *Circulation*, vol. 68, no. 3, pp. 470–476, 1983.

[47] "Farlex partner medical dictionary," http://medical-dictionary.thefreedictionary.com/electrocardiogram, 2012, accessed at 1 May 2015.

[48] "Married2medicine web page," http://married2medicine.hubpages.com/hub/Special-Investigations-In-Cardiology-Radiology-And-Electrocardiography-ECG, 2012, accessed at 10 June 2015.

[49] J. Rydzek and Z. Gasior, "The importance of resting ecg in diagnosis of coronary heart disease exacerbation and in heart rate disregulations in patients under home palliative care," *Wiadomosci lekarskie (Warsaw, Poland: 1960)*, vol. 64, no. 1, pp. 56–62, 2010.

[50] H. Delfs and H. Knebl, *Introduction to cryptography: principles and applications*. Springer Science & Business Media, 2007.

[51] H. Feistel, "Cryptography and computer privacy," *Scientific american*, vol. 228, pp. 15–23, 1973.

[52] W. Diffie and M. E. Hellman, "Special feature exhaustive cryptanalysis of the nbs data encryption standard," *Computer*, vol. 10, no. 6, pp. 74–84, 1977.

[53] N.-F. Standard, "Announcing the advanced encryption standard (aes)," *Federal Information Processing Standards Publication*, vol. 197, 2001.

[54] W. Diffie and M. E. Hellman, "New directions in cryptography," *Information Theory, IEEE Transactions on*, vol. 22, no. 6, pp. 644–654, 1976.

[55] K. S. McCurley, "The discrete logarithm problem," in *Proc. of Symp. in Applied Math*, vol. 42, 1990, pp. 49–74.

[56] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of computation*, vol. 48, no. 177, pp. 203–209, 1987.

[57] Y. Desmedt, "Man-in-the-middle attack," in *Encyclopedia of Cryptography and Security*. Springer, 2011, pp. 759–759.

[58] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[59] E. Bach, *Algorithmic Number Theory: Efficient Algorithms*. MIT press, 1996, vol. 1.

[60] S. Zheng, D. Jiang, and Q. Liu, "A role and activity based access control model for university identity and access management system," in *Information Assurance and Security, 2009. IAS'09. Fifth International Conference on*, vol. 2. IEEE, 2009, pp. 487–490.

[61] I. O. for Standardization, *Information processing systems-open systems interconnection-LOTOS-a formal description technique based on the temporal ordering of observational behaviour*. ISO, 1989.

[62] P. Gæde, P. Vedel, N. Larsen, G. V. Jensen, H.-H. Parving, and O. Pedersen, "Multifactorial intervention and cardiovascular disease in patients with type 2 diabetes," *New England Journal of Medicine*, vol. 348, no. 5, pp. 383–393, 2003.

[63] E. D. Peterson, M. T. Roe, J. S. Rumsfeld, R. E. Shaw, R. G. Brindis, G. C. Fonarow, and C. P. Cannon, "A call to action (acute coronary treatment and intervention outcomes network) a national effort to promote timely clinical feedback and support continuous quality improvement for acute myocardial infarction," *Circulation: Cardiovascular Quality and Outcomes*, vol. 2, no. 5, pp. 491–499, 2009.

[64] G. Demiris, L. B. Afrin, S. Speedie, K. L. Courtney, M. Sondhi, V. Vimarlund, C. Lovis, W. Goossen, and C. Lynch, "Patient-centered applications: use of information technology to promote disease management and wellness. a white paper by the amia knowledge in motion working group," *Journal of the American Medical Informatics Association*, vol. 15, no. 1, pp. 8–13, 2008.

[65] B. C. Neuman and T. Ts'o, "Kerberos: An authentication service for computer networks," *Communications Magazine, IEEE*, vol. 32, no. 9, pp. 33–38, 1994.

[66] J. Fraden and M. Neuman, "Qrs wave detection," *Medical and Biological Engineering and computing*, vol. 18, no. 2, pp. 125–132, 1980.

[67] D. Kreiseler and R. Bousseliot, "Automatisierte EKG-Auswertung mit Hilfe der EKG-Signaldatenbank CARDIODAT der PTB," *Biomedizinische Technik/Biomedical Engineering*, vol. 40, no. 1, pp. 319–320, 2009.

[68] W. Feller, *An introduction to probability theory and its applications*. John Wiley & Sons, 2008, vol. 2.

[69] D. Karaoğlan and A. Levi, "A survey on the development of security mechanisms for body area networks," *The Computer Journal*, vol. 57, no. 10, pp. 1484–1512, 2014.