PHILIPPINE ENGINEERING JOURNAL
PEJ 2017; Vol. 38, No. 1: 27-38

A Design for Task-Role Based Access Control for Personal Health Record Systems

Rose Ann S. Zuniga[1] and Susan P. Festin[2]
Computer Security Group, Department of Computer Science
University of the Philippines - Diliman
Quezon City, Philippines
[1]rsalezuniga@gmail.com ; [2]sfestin.up@gmail.com

Abstract – *We present our design for a Task-Role Based Access control system for Personal Health Records (PHR). Current access control models deployed for PHR systems are, at best, based on role-based models. This neither allow for flexibility nor fine-grained restrictions on access to records. The ideal situation is to have a dynamic, task-based access control model on top of the role-based restrictions. Multiple constrains were also added to provide a more fine-grained access. Furthermore, specific policies for PHR systems were also defined. From our survey of existing PHR systems none provide these combination of dynamic access control coupled with constraints and roles. We implemented a prototype, a hybrid PHR-EMR (Electronic Medical Record) system, of our design where we applied the security model we are proposing. We also conducted a usability testing and our evaluation shows that our design can be used and implemented in an actual PHR.*

Keywords*: Access Control, TRBAC, Task – Role Based Access Control, PHR, Personal Health Record System.*

I. INTRODUCTION

The International Organization for Standardization (ISO) identified access control as an important safety service level to protect digital information. Access control can be defined as the process that determines whether a certain request of access in a system will be granted. Permission of access can be decided upon based on the owner of the data (Discretionary Access Control), based on the decisions of a central authority (Mandatory Access Control), based on the tasks to be performed (Task based) or based on the role of the requester (Role based). Access control aims to limit the behavior and operation of a system user and to prevent illegal user from intruding and harming the system and its users.

Making access control more fine-grained has been the goal of many studies that range from improving existing models by adding more constraints to integrating them to other access control models. The current trend is mainly on Role-Based Access Control (RBAC). The general notion for RBAC is that there are permissions granted to roles and users are assigned to appropriate roles. A user playing a role is allowed to execute all accesses for which the role is authorized. In general, a user can take different roles on different occasions, and several users can play the same role simultaneously [1]. Among RBAC's advantages, according to [1], are Authorization Management, Hierarchical Roles, Least Privilege, Separation of Duties, and Constraints Enforcement. However, there are other real world scenarios that are not addressed. For instance, there are relationships that can occur that simple hierarchical relationship may not cover. For example, a nurse may need to be allowed to do specific task on behalf of the doctor she is assisting, but neither role is a specialization of the other. Furthermore, since the role identifies the privilege that one may execute, one would think that the identity of the user

is not important. But there are scenarios where the requestor's identity needs to be considered even when a role-based policy is adopted [1]. For instance, a doctor may be allowed to give and write diagnosis but he may be restricted to his own patients only.

In the Task-Based Access Control (TBAC), permission is assigned to tasks and users can only obtain the permission during the execution of the task. It is not static and invariable, it can change along with the context of the task, and it provides dynamic, real-time safe management during task processing [2]. One of TBAC's limitations is its primitive specification of complex policies, management, delegation, and revocation of authorization. A more fine-grained component to support TBAC is needed [3].

Although studies on the combination of RBAC with other access control models (e.g. Temporal-Role-Based Access Control, Personalize Access Control) already exist, only few have studied and tried its actual implementation.

For this study, the focus on information security is on designing an integration of RBAC and TBAC or the Task-Role-Based Access Control (TRBAC) and its implementation on further securing medical record, specifically, PHR Systems. There is a current trend on electronic health record researches due to the enactment of the Health Insurance Portability and Accountability Act (HIPAA) in the US, Personal Health Information Protection Act (PHIPA) in Ontario, Canada, and other similar health act in other countries, such as UK and Germany. Many researches have focused their studies on providing systems that helped institutions go electronic and on applications that helped automate any health related activities/tasks on different health institutions.

The American Medical Informatics Association and Markle Foundation define PHR as "an electronic application through which individuals can access, manage and share their health information and that of others for whom they are authorized in private, secure, and confidential environment" [4], [5]. PHR Systems aim to make a patient more involved and conscious about his own health. It is mainly controlled and managed by the patient but it would still involve multiple resources and multiple users who perform different tasks.

Under the Security Rule of HIPAA is the Technical Standards. It includes Access Control, which has the following implementation specification: Unique User Identification, Emergency Access Procedure, Automatic Logoff, and Encryption and Decryption. The first two are required, which means they are part of the minimum standard while the last two are addressable standards or optional standards that may or may not be implemented.

For this study, the features of RBAC and TBAC were used as framework of the access control for the PHR System, additionally; multi-constraints were applied for additional security. Furthermore, specific policies for PHR system were defined to address the needs of a secure PHR system.

This paper is organized as follows. We will provide a theoretical framework on TRBAC on Section 2. System Design and Access Control will be discussed in Sections 3 and 4. We will discuss the implementation and the testing results on Sections 3 and 4. Conclusions are presented in Section 7.

II. TRBAC

Task-Role-Based Access Control (TRBAC) is an access control model developed to put constraints on the tasks and the corresponding roles of those who will try to access a system. Below is the framework of the TRBAC model.

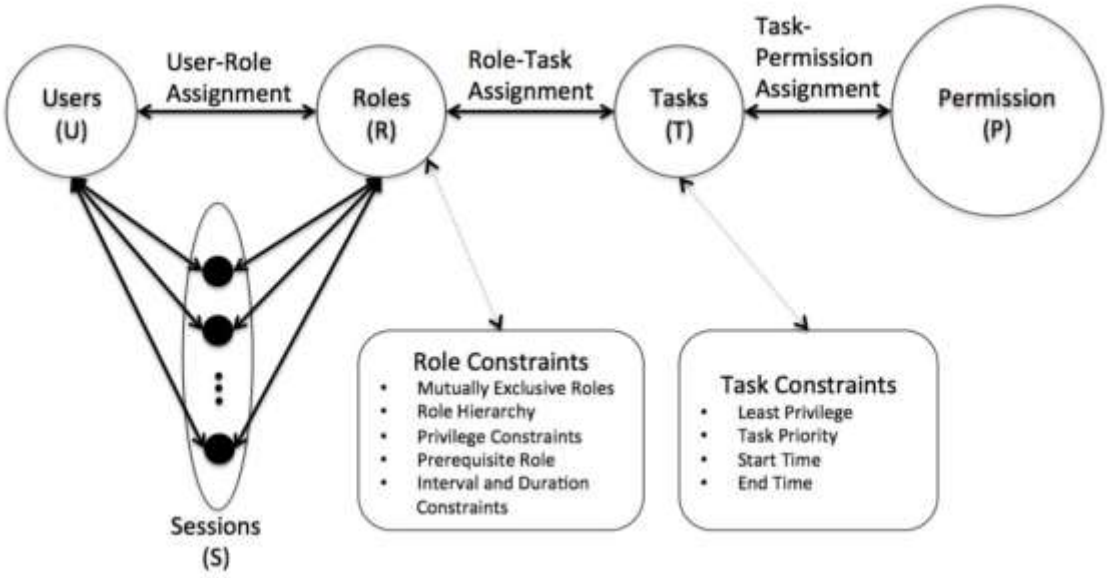

Figure 1. Task - Role Based Access Control Model

The user with assigned role or roles would activate some of those roles through a session. Tasks are assigned to users via their role/roles in the system. A user's permission to access certain files is determined by the tasks assigned to him. Constraints are important aspect of access control and are a powerful mechanism for laying out higher-level organization policy. With constraints, we would be able to address some issues that RBAC and TBAC models left open. The following are the constraints added [7].

1. *Role Constraints*

- Mutually Exclusive Roles: The same user can be assigned to at most one role in mutually exclusive set.
- Role Hierarchy: A role R_1 possesses all permissions of role R_2 if role R_1 inherits role R_2.
- Privilege Constraint: The maximum privilege possessed by a role. The role defines the maximum capability a person can do.
- Prerequisite Role: A user can be assigned to role R_1 only if the user is already a member of the role R_2 (R_2 is a prerequisite of R_1).
- Interval and Duration: The interval constraint denotes a single or a set of intervals during which the corresponding role enabling or activation event can occur. The duration constraint is used to specify the duration for which enabling, assignment, or activation of role is valid.

2. *Task Constraints*

- Least Privilege: The necessary permission authorized to a user should be minimized.
- Task Priority: Task with the highest priority should be the first to be executed.
- Start Time: The execution of a task must begin at the stipulated time, or it will fail automatically.
- End Time: The execution of a task must end at the stipulated time, or it will fail automatically.

III. SYSTEM DESIGN

In the ideal patient-centric PHR System model, patient data is accessible not only to the primary health provider but also to other users who might help in maintaining the wellbeing of a patient. In such a system, not only the people who provide primary health care are involved and can have access to the patient's record. Apart from the doctors, the nurses and other medical practitioners, other users who can also be given access are family members, schools or employers, pharmaceutical companies, insurance companies, and even researchers and friends. However, since the described system is complex, the study will focus on the primary health provider only. For this study, the role would be limited to Doctor (primary and consultant), and Nurse (may include medical technicians, radiologists, etc.) and Patient.

A patient may consult many doctors affiliated in different hospitals or institutions or specializing in different aspects of health. However, the patient can only choose one primary doctor and the rest would be consultant doctors. The primary doctor may consult another doctor or may refer the patient to another doctor, that doctor's role would be consultant doctor. A consultant doctor could be a primary doctor of another patient.

Figure 2 shows a high level design of our Access Control Protocol. In the design, all access control information about the user is stored in a web server. In order to gain access to the contents of the database, a user must first be authenticated.

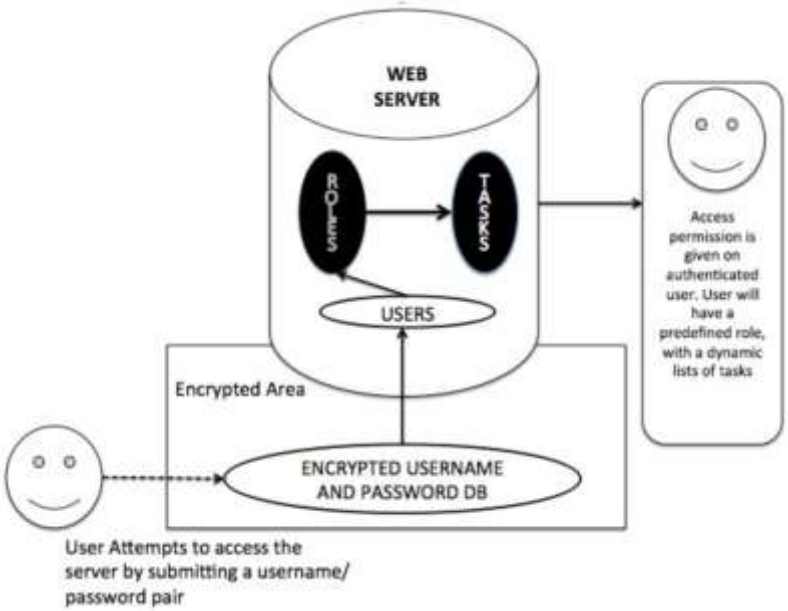

Figure 2. High Level Design

After a user activated a role upon login (User-Role Assignment), a dynamic task list associated to the role will be given (Role-Task Assignment). It is important to note that the task list is constantly changing depending on the given task provided by the assigning authority. On task creation, the assigning authority determines the privilege needed to execute the task (Task-Permission Assignment).

IV. ACCESS CONTROL POLICIES FOR A PHR SYSTEM

To provide a fine-grained access control over a patient data, policies summarized in the following tables (Tables 1 and 2) are proposed:

Table 1. Granting and Revocation of privileges

	Primary Doctor		Consultant Doctor		Nurse	
	Grant Privileges	Revoke Privileges	Grant Privileges	Revoke Privileges	Grant Privileges	Revoke Privileges
Patient	χ	χ	χ	χ		
Primary Doctor			χ	χ	χ	χ
Consultant Doctor					χ	χ
Nurse						

The first layer of access control is the granting/revocation of privileges. In the policy discussed above, we can see that a patient will have the power to give/revoke privilege to any doctor, yet he can't assign privilege to a nurse. The primary doctor can assign privilege to a consultant doctor or to a nurse (if applicable). The doctors give and revoke nurse access.

Table 2. Task Privileges

	Create	Read	Edit	Archive
Patient	χ	χ	⁄	⁄
Primary Doctor	χ	χ	χ	⁄
Consultant Doctor	χ	χ	⁄	⁄
Nurse	⁄	⁄	⁄	⁄

Legend: χ = full access; ⁄ = partial access

The patient and the consultant doctor have full *create* and *read* privileges, but they can only *edit* and *archive* entries that they created. The primary doctor has full *create, read* and *edit* privileges but limited *archive* privileges. The nurse on the other hand, has limited actions. This depends on two things:

(1) whether he was the one who created the entry that he wants to *read*, *edit* or *archive* or, (2) whether the supervising doctor has provided him with such privileges.

Temporal properties of privileges include the following: (1) A consultant doctor's privileges over a patient's data are valid only while he is listed as the patient's consultant doctor. (2) A nurse's privileges over a patient's data are valid only while he is listed as a nurse assigned to the patient. (3) A nurse's privileges are valid only while he is on-duty.

V. IMPLEMENTATION

As part of this research, we also developed a prototype implementation of our PHR access control design. The implementation is web-based and can be access through a browser. We developed the prototype using PHP, with Code Igniter as framework and MySQL for database.

Patient view includes form entries, reading, editing and archiving. Forms included in the prototype are: Allergies, Medications, Consultation Form (SOAP format) and the General Health Assessment Form that patients answer prior to doctor's appointment. Patient profile includes personal information, emergency contact, and insurance information. Through his medical team view, he can view all the doctors and nurses who have access to his data, and he can grant and revoke privileges.

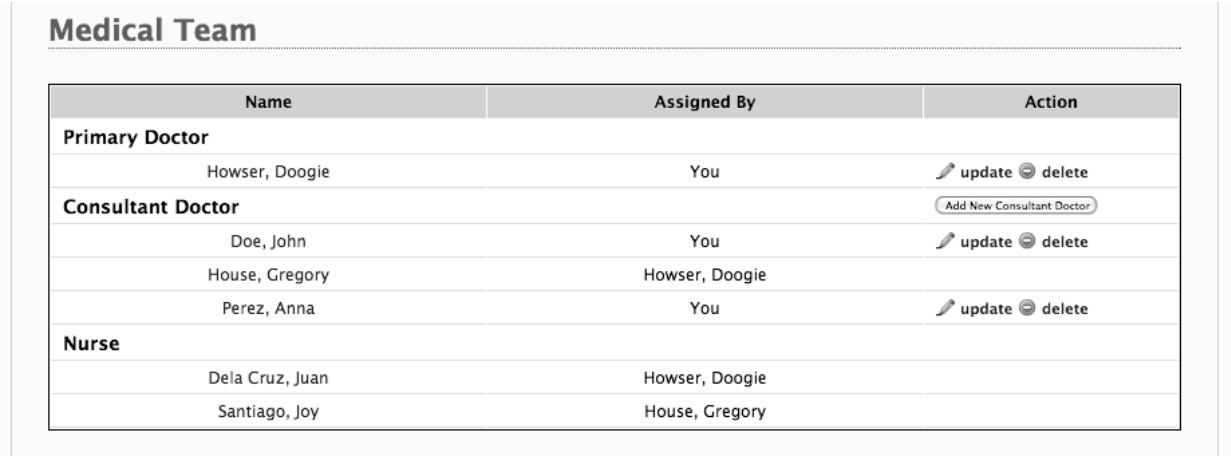

Figure 3. Patient's view: Medical team

Doctor's view, in addition to the profile management and form management, includes a list of all of their patients, which also indicates his secondary role (primary or consultant). There is also a module to connect a patient to another user (another doctor or nurse), and most importantly, assign task module, Fig.4. In the assign task module, the doctor can assign tasks to a Nurse or another Doctor. It has an option to set time to ensure that no task will be left pending for a long time. A task prioritization field is also placed since the system should be able to flag down important tasks that are needed as soon as possible.

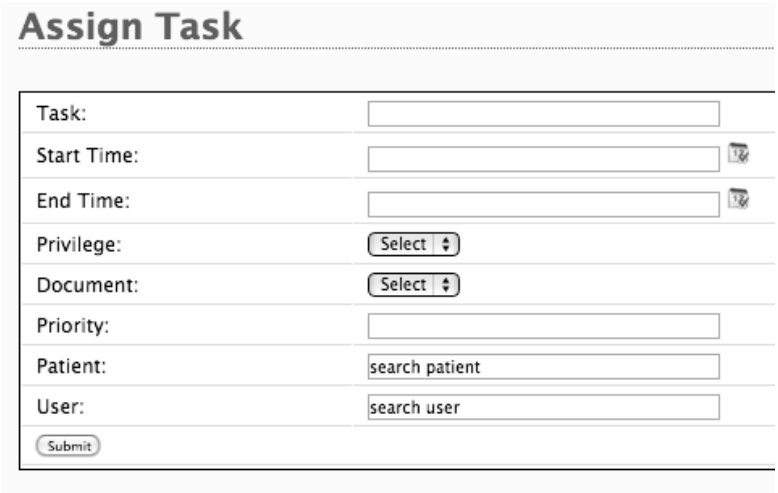

Figure 4. Doctor's view: Assign Task

Upon login, the first thing that the nurse will see is the list of tasks assigned to him. Here, the nurse would be able to see which patient and what privilege is given to him, and other important things such as the priority and the time duration. If the task is completed, the nurse can mark it done. When the task is marked done or when the time duration elapse, the nurse will not be able to access the files indicated in that task. If the nurse is off-duty, he will not be able to access any patient data and he will not be able to mark a task as done.

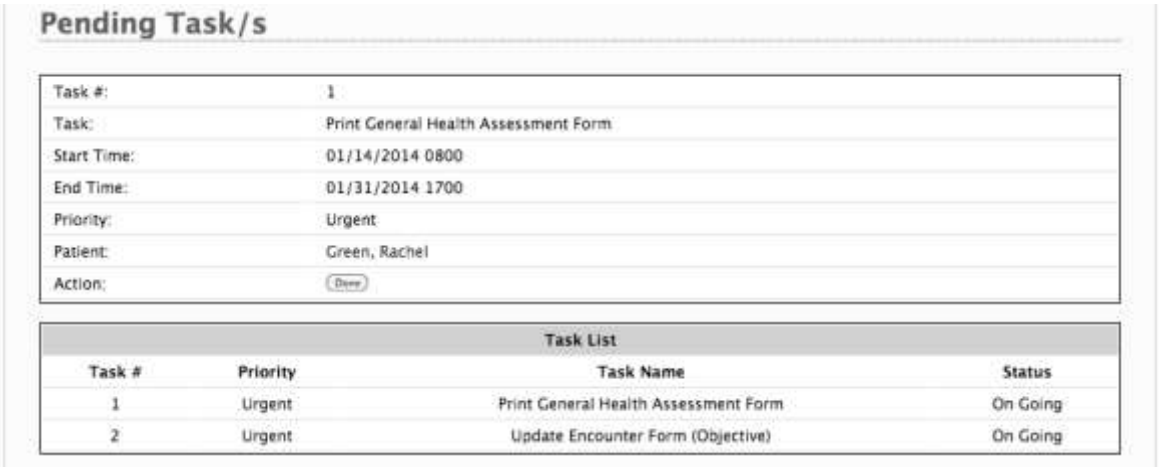

Figure 5. Nurse's view: Task list

The following table shows that our proposed access control protocol and the implementation satisfy all the role and task constraints that we presented.

Table 3. Constraints Implementation

Role Constraints	
Mutually Exclusive Roles	The system is designed such that upon login, a user can only be assigned with a "Patient", "Doctor" or "Nurse" role.
Role Hierarchy	In our design, a Primary Doctor and a Consultant Doctor inherits all the properties of a Doctor role.
Privilege Constraints	These privileges are specified in the role policy that we described above.
Prerequisite Role	In our design, a user can be assigned as a Primary or Consultant Doctor if he is initially assigned as a Doctor.
Interval and Duration Constraint	Some roles (i.e. Consultant Doctor/Nurses) have specific validity duration as defined by the assigning authority.
Task Constraints	
Least Privilege	Each task comes with an optimal number of privileges needed for the user to function.
Task Priority	This constraint is present in our proposed scheme; it is a variable that the user can set when creating tasks. Higher priority will appear on the top most of the list.
Start Time	On task creation, a variable for start time is available. This will also set the time where the user can access the specified document.
End Time	Another variable that needs to be set.

VI. SAMPLE WORKFLOW

Figure 6 shows our sample workflow diagram. The sample workflow starts when the patient starts feeling some pregnant-like symptoms like headache, vomiting and missing her monthly menstruation. As such, the patient will visit her General Physician, who in our model will be her Primary Doctor (P.D). The P.D. will request for access to the patient if he still has no access. After gaining access, the P.D will give access to the nurse. The nurse will now conduct more tests to the patient and he will record the results of the tests to the patient's PHR. After receiving the reports, the P.D. will now interpret them and decide if the patient should be forwarded to a different consultant doctor.

If the results of the test shows that the patient is likely pregnant (i.e. HCG>1.0), then he will recommend the patient to a consultant doctor (C.D), in this case an OB-Gynecologist. He can also give access to the C.D. The patient will now visit the C.D. and have further tests (i.e. vaginal ultrasound). The C.D. will decide if the patient is pregnant or not depending on the result of the tests.

If the patient is NOT pregnant, then she will be released, on the other hand, if the patient is pregnant, then she will be regularly consulting with the OB-Gyne. After being diagnosed as pregnant, she will also be tested for possible red flags such as hypertension or diabetes. If the tests show that the patient has hypertension, then she will be recommended to another C.D. (cardiologist). This time the

OB-Gyne will be the one to recommend and give the C.D. access to the program. The cardiologist will test if hypertension is pregnancy related or not. The diagnosis will be forwarded to the OB-Gyne for proper maintenance.

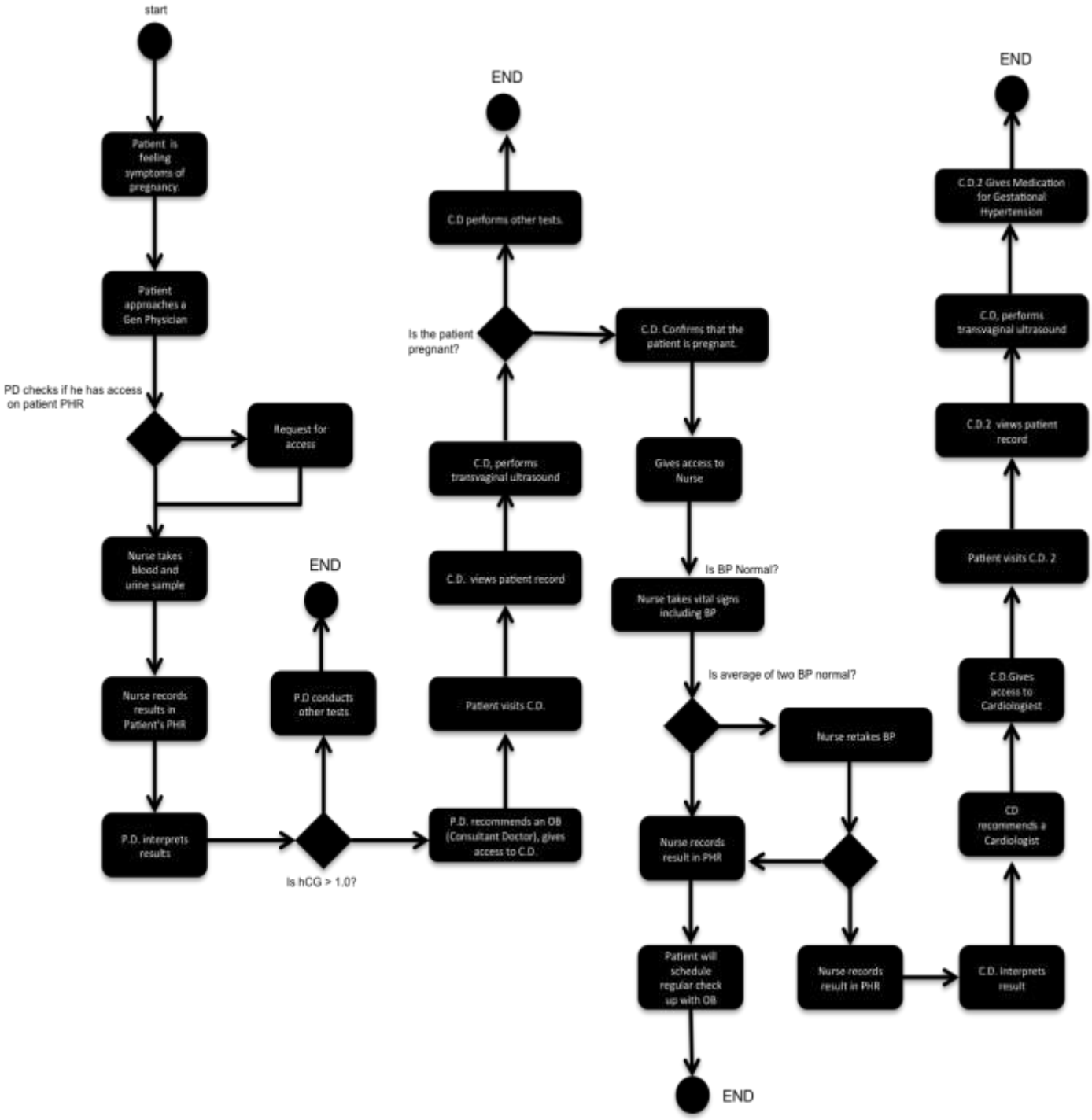

Figure 6: Sample Workflow

In our sample workflow, we should be able to test the following security test cases:

1) Patient gives PHR access to his Primary Doctor.
2) Primary Doctor gives PHR access to nurse, but the privileges will depend if the nurse is on duty or not.
3) Primary Doctor gives limited access to Consultant Doctor. Though the C.D can view the patient's PHR, he has limited write and archive privileges.
4) The Consultant Doctor gives limited access to another Consultant Doctor.

VII. TESTING

The prototype was subjected to a usability and functional test to determine if the system, though still a prototype can be extended to cover a whole PHR. It was tested in a health center division of the Manila Health Department. Three doctors from different fields joined us together with 3 staffs that played as nurses and 5 patients. The number of usability testers is small as suggested by Nielsen in [6]. According to him, the best results come from testing no more than 5 users and running small tests as you can.

The test conducted was done to show that: (1) The system is usable, (2) The amount of security used does not limit the usability of the system, (3) the features are working, and lastly (4) It exhibits the different properties and constraints presented in the design.

To quantify the results, we asked the testers to fill up a survey. Questions are answerable using the following ratings: (5 points) Strongly Agree, (4 points) Agree, (3 points) Indifferent, (2 points) Disagree, (1 point) Strongly Disagree, and Not Applicable.

Initially, we let the users answer a demographic questionnaire as pre-test to gather personal information from each individual user. For the actual testing, we already assigned roles (e.g. who will play as primary or consultant doctor for whom) for each user and we let them follow a workflow using a manual. The manual has exercises to be carried out; it allows the users to test the restrictions of TRBAC and to explore the features of the system.

The patient's main task is to maintain the information in his PHR and to assign a primary doctor. The doctor, who may be the primary or consultant doctor of some of the patient testers, must associate a patient to his assigned nurse and to refer a consultant doctor. Another important task of the doctor is to assign tasks to his nurse. The nurse should be able to execute the task and mark it done after.

All users gave a positive mark on the difficulty of creating an account. The patients are divided in the difficulty of creating new forms with 2 of them agreed, 2 disagreed and 1 indifferent. However, high remarks are given for editing and archiving documents. All doctors and nurses said that creating documents is easy, and 100% strongly agreed that editing and archiving is easy. We asked the doctors to create tasks and all of them (2 strongly agree, 1 agree) said that creating tasks is easy, the same positive result was gathered on linking patients to a consultant doctor or nurse. All nurses agreed that after a task's access expires (off-duty or beyond end time), they can no longer access the document specified on the task. Based on their responses in the manual's exercises, all features of the prototype are working

(no bugs or error were encountered and the system behaved as it is intended) and none of them were able to access any of the data that they are not authorized to access.

100% of users agreed (7 strongly agreed) that limiting the access of some of the users based on the task assigned to them further secures the data. Moreover, 100% would promote the use of PHR systems since they all think that it is useful, helpful and needed. Overall, the PHR system prototype received a high mark with an average of 4.67, while TRBAC received an average mark of 4.44.

VIII. CONCLUSIONS

We presented in this research work a novel, effective, secure and useful protocol on the management of Personal Health Records. Despite the additional overhead due to additional layers of security, changes in workflow due to the security protocol and the need to provide additional security tokens or additional security modules to existing systems, it is important to note that overall, our proposed scheme's benefits will outweigh the additional overhead. The work has shown that we are able to achieve our claim that the design proposed is novel, dynamic and useful. The following are our detailed conclusions.

- The proposed scheme is novel. Based from our exhaustive survey of PHR's, this is the only system that applies Task – Role Based Access Control System in a PHR.

- The proposed scheme is functional, as all functions required in a standard PHR is still present in this system. We conducted a functional test of the system to test whether the required functions do exists, and are functioning.

- The proposed scheme is more dynamic as compared to existing PHR's. It addresses some of the HIPAA technical standards as shown in the table below. Furthermore, it used a Task and Role based Access Control System. The TRBAC scheme answers the limitations presented by other schemes used by other PHR's. The TRBAC provides a list of roles that can be assumed by a user upon login and there is a dynamic task list depending on the assigning authority.

- The proposed scheme is useful. This result is obtained when we asked a group of potential users to test a prototype given our sample workflow. The result of the survey and test shows that this scheme is usable in common workplace scenario.

Table 4. HIPAA Technical Standards Implementation

Standards	**Implementation Specifications**	**Delivery**	**In our design**
Access Control	Unique User Identification	Required	Implemented
	Emergency Access Procedure	Required	Implemented
	Automatic Logoff	Addressable	Implemented
	Encryption and Decryption	Addressable	Implemented
Audit Controls		Required	Implemented
Integrity	Mechanism to Authenticate Electronic PHI	Addressable	Beyond the scope
Person or Entity Authentication		Required	Implemented
Transmission Security	Integrity Controls	Addressable	Implemented
	Encryption	Addressable	Implemented

IX. REFERENCES

[1] P. Samarati, and S. D. di Vimercati, S. D.,Access Control: Policies, Models, and Mechanisms in FOSAD '00 Revised versions of lectures given during the IFIP WG 1.7 International School on Foundations of Security Analysis and Design on Foundations of Security Analysis and Design: Tutorial Lectures, UK: Springer-Verlag London, 2001 pp.137-196

[2] C.-x. Zhang, Y.-x. Hu, and G.-b Zhang. Task-Role Based Dual System Access Control Model. International Journal of Computer Science and Network Security, Vol.6 No.7B, 2006, pp. 211-215.

[3] Mohammad, T. Khdour, G. Kanaan and R. Kanaan, Analysis of Existing Access Control Models from Web Services Applications' Perspective, in Journal of Computing, Vol. 3 No.3, 2011, pp.10-16.

[4] K. B. Johnson, Project HealthDesign: Advancing the vision of consumer-clinician-computer collaborations, in Journal of Biomedical Informatics, Vol. 43 No. 5, 2010, S1-S2.

[5] Ogbuji, K. Gomadam and C. Petrie, Web Technology and Architecture for Personal Health Records, in Internet Computing, IEEE , Vol. 15 No. 4, 2011, pp.10-13.

[6] J. Nielsen, Why You Only Need to Test with 5 Users, 2000. Retrieved January 30, 2014 from http://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/

[7] L. Yao, X.Kong and Z. Yu, A Task – Role Based Access Control Model with Multi Constraints, International Conference on Network Computing and Advanced Information Managament, 2008