A Hybrid Web Based Personal Health Record System Shielded with Comprehensive Security

Jennifer Israelson
University of North Texas
Department of Computer Science Denton, TX USA
jennifer.israelson@gmail.com

Ebru Celikel Cankaya
University of North Texas
Department of Computer Science Denton, TX USA
ebru.cankaya@unt.edu

Abstract

We present the design and development of a hybrid, web-based scheme for creating, maintaining and sharing personal health records (PHRs) with embedded security. We adopt a hybrid approach to processing PHRs and present a prototype called Personal Health Manager (PHM) that is based on this hybrid model. PHRs in the PHM prototype are owned by patients but updated by medical professionals who have been granted access to the record by the owner. Our prototype design provides a framework to begin exploring the major security concerns such as confidentiality, integrity, availability, authentication, authorization and non-repudiation (as part of the X.800 security architecture) that arise with the adoption of an electronic method for maintaining and sharing highly sensitive healthcare information. We provide comparison of our work with existing PHR tools based on parameters that distinguish PHR models such as ownership of data and security considerations.

1. Introduction

The current environment in the United States, which includes both the need to reduce healthcare costs and the prevalence of electronic sources (including web-based and mobile), encourages the development of electronic methods for storage and sharing of health-related information. The advent of the electronic personal health record (PHR) brings many advantages, including the potential for reduced health care costs and better health, but also many concerns, primarily those focused on the privacy of health-related information [3]. With the advent and rapid growth use of Internet and mobile technologies, accessing and processing data remotely is part of our lives today. Unsurprisingly, we have our health records stored and accessed on digital media, so that we have fast, secure, and efficient processing of one of the most important assets of our lives. PHRs are used primarily by patients, as well as healthcare providers, and policymakers to improve the quality, effectiveness and cost of healthcare [1]. PHR is an element of the broader Electronic Health Record (EHR), which is controlled by the healthcare provider and has a broader aspect [2].

The health care system in the United States is undergoing immense changes after the passage of H.R. 3962, the Affordable Health Care for America Act, in March 2010. The new law focuses on reducing health care costs using a variety of methods. At the same time, prevalence of computers and individuals’ access to computers makes the idea of an online personal health record (PHR) system a desirable and viable goal.

The definition of PHR is evolving, but a generally accepted definition is an online record of an individual’s health history. This record may include illnesses and hospitalizations, medications and dosages, allergies and drug reactions, family history, chronic conditions, immunization records, and imaging and lab results. There are many potential benefits for both patients and their caregivers, for healthcare providers and payers, and for employers to maintaining PHRs [3]. For the patient, the benefits are many and include help in the support of wellness activities, improved understanding of health issues and the ability to verify the accuracy of records and improved communication with healthcare providers. Healthcare providers also benefit from improved communication and documentation of that communication, with the patient and through increased knowledge of potential drug interaction (especially when the patient sees multiple doctors), as well as from the increased ability to provide patients with access to information or services.

Both payers and employers benefit from better wellness and preventive care. PHRs can assist in the portability of records across plans and can even improve workforce productivity through better health and easier access to records. A report on PHRs by the National Committee on Vital and Health Statistics [3] notes these advantages. The same report makes recommendations in areas critical to the implementation of a personal health record system, and key among these is the area of security. Security of PHR systems and the information stored by those systems is crucial to the implementation and
acceptance of those systems. The report recommends that standards be developed regarding authentication, access control, authorization and auditability.

This study presents the design and development of a fully functional, potentially integrated PHR system that takes into account the fundamental security requirements: confidentiality, integrity, authentication, authorization, availability, and non-repudiation.

The rest of the paper is organized as follows: In section 2, we investigate the background and related work. Section 3 presents various implementations of PHR systems, and extensions we consider to expand existing systems as well as a description of the PHM prototype. We provide a detailed explanation on experimental work and results on Section 4. Section 5 concludes the paper and presents our intentions to further extend our work in the future.

2. Background and related work

In spite of initial apprehension about whether PHRs will receive recognition - primarily because of inexperienced, prejudiced patients with security concerns - PHRs have gone a long way. A survey conducted by [4] on 330 patients in year 2000 revealed that patients had varying levels of sympathy towards the adoption of PHRs with not too promising results: their level of support varied widely based on their demographic properties, as well as their concerns about reliability, trust issues on shared media – especially on Internet -, and the belief that the PHR tool that was introduced to them was technically difficult to use.

[5] estimated a total outpatient savings of $20.4 billion and a total inpatient savings of $57.1 billion by the year 2020 with a 90% adoption rate of PHR. This enormous potential savings make PHR systems a very attractive and profitable application. The same study suggests that PHRs will lessen adverse drug instances, improve the effectiveness of near term as well as long-term preventive care and chronic disease management. Despite high promising return on investment (ROI) figures – measured in terms of better quality of service, patient and cost tracking, documentation, per hour shifting requirements for healthcare staff, lessened requirement to repeat expensive tests, rather than on financial terms [6], only 17% of healthcare professionals and 10% of hospitals are using PHRs yet [7, 8]. Also, it would give harm rather than good if government policies on healthcare are not adopted [9], and security issues are not handled properly [10].

In literature, various research exists measuring the implications of adopting PHRs. In [11], the authors present the result of three case studies between years 2000 through 2007: MyChart PHR application at Palo Alto Medical Foundation, PatientSite PHR application at Beth Israel Deaconess Medical Center, and Indivo PHR application at Children’s Hospital Boston. All three cases resulted in wide adoption of aforementioned applications and yielded valuable insights on present and foreseeable challenges.

There are three models for implementing PHRs [12], namely standalone (on individual PCs), integrated (associated with many medical providers or their own electronic health records (EHRs), policy providers, pharmacies, etc.), and tethered (an integrated PHR that is restricted to one EHR and policy provider) [12]. The decision to choose one method depends on many factors such as people and organization, technical infrastructure, and future considerations [13]. Many examples of each model can be found online, with even free demos, trial versions. For example, Google Health [14] for a stand alone PHR, Health Record Bank [28] for an integrated PHR, and Aetna with a comprehensive demo [16] of a tethered PHR.

Our PHM design is based on a hybrid model that is most similar to the integrated model with multiple providers and components. Though, it aligns most closely with the integrated PHR model, our design possesses two important distinctions. First, the patient maintains ownership of the PHR but only authorized medical professionals may change it. Second, the model incorporates a review process in which both the patient and medical providers participate. These additions to the model provide extra control over sensitive data and ensure that the integrity of the data is maintained.

Other recent developments can be found in diverse areas of application of PHRs, such as MyHealthVet [17] designed specifically providing PHR solutions for veterans, USB-based PHRs such as MedicTag, LifeSaver, and many others that provide USB-based commercial solutions for patients [18], and even semantic based PHR applications involving agents, such as TrialX, which extracts PHR records to match most relevant clinical trials, i.e. research conducted on human beings to observe the effects of new drugs [19].

Nowadays, with the advances in technology and lowering costs in the underlying infrastructure, automated healthcare service is an emerging topic. In fact, the University of Illinois at Urbana-Champaign is leading a collaborative effort funded by the Office of the National Coordinator for Health Information Technology, a subsidiary of the U.S. Department of Health and Human Services [20] to study security
concerns involved in electronic health records, health information exchange and telemedicine. This $60 million effort (one of the four parts focuses on security) is intended to develop security and privacy protocols and to establish a security/healthcare-minded community that will continue to focus on these problems after the study ends.

Our work, to the best of our knowledge, differentiates from its peers by introducing the novel idea of providing a PHR that is owned by the patient and is updated by the healthcare provider and by the integration of a review process that includes the patient and the medical providers. The incorporation of these changes is intended to lead to a higher level of integrity in the data stored in the personal health record.

3. Personal health record implementations

3.1 Existing personal health record implementations

According to a report in [21], two types of products exist in the market today: One is a Personal Health Record (PHR) – an individual owns this record and updates it himself or has it updated for him by a third party that contacts medical providers on his behalf. Google Health, PassportMD and Health Trio are mentioned as three of the leading products. The second type is an Electronic Health Record (EHR), sometimes referred to as an Electronic Health/Medical Record (EH/MR), that is owned and controlled by a health care provider or plan. Health Record Banking [28] and Aetna’s [16] Other sources [12] differentiate among standalone, integrated and tethered PHR systems. We compare these three PHR classifications and add a fourth, the hybrid model, on which our Personal Health Management prototype is based.

The standalone model is owned and updated, usually manually, by the patient. It may contain data from any medical provider, but the patient enters the information. The primary drawbacks are the manual data entry and the uncertainty of providers about the accuracy of the data [12].

There is no single ownership for the integrated PHR. Instead, it is owned and updated partially by the patient, as well as by insurance companies, pharmacies, and healthcare providers [12]. (In that sense, our design takes on the characteristics of the integrated PHR scheme because any healthcare provider, insurance company or pharmacy can update it.) The integrated PHR offers the most advantages of the first three models [12] in that it allows consumers access to provider records, offers a potentially comprehensive view of a patient’s health, and may be viewed as more reliable than standalone PHRs because data has come from a medical provider.

The tethered model is the most restrictive model among the others. It is a form of the integrated model, but it is tightly coupled with a single medical provider or insurance company. The provider controls patient data, although some allow limited user input [12]. The obvious limitation to this plan is that it does not provide a complete picture of a patient’s health, only a window to information for which that particular provider is responsible.

We summarize the properties of each of these models in Table 1.

<table>
<thead>
<tr>
<th>PHR model comparison</th>
<th>PHR updated by</th>
<th>PHR owned solely by provider</th>
<th>Contains data from multiple providers</th>
<th>Changes made by medical provider</th>
<th>Portable across IP/MPs</th>
<th>Integrated review process that includes P, MP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standalone [12]</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Integrated [12]</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tethered [12]</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hybrid</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>X (planned)</td>
</tr>
</tbody>
</table>

As the previous discussion and Table 1 illustrate, each of the current models has its advantages. We question, however, whether we have yet arrived at the model and propose a hybrid model that seeks to combine the advantages of all existing models.

We see an advantage to keeping the ownership of the PHR with the patient. The patient is the only person familiar with all aspects of and influences on his health and is, of course, the one to benefit the most from detailed, complete and accurate medical records. Although a medical professional, because of his training, may more likely to enter accurate data, that medical professional may not be as familiar with a patient’s entire medical picture as the patient himself.

We propose a hybrid model that leaves the ownership of the record with the patient (as in a standalone model), allows updates by only medical professionals (as in the integrated model) and that incorporates an integrated review process through which the patient may review his record and submit change requests, as needed, to the medical
3.2 Description of personal health management application

Personal Health Management (PHM) is a secure, web-based application that implements a system for maintaining personal health records (PHRs). The PHM system is designed for an imaginary company, whose mission is to provide a framework for patients to maintain PHRs and for medical providers to update them in a secure environment.

The personal health record includes basic information about the individual (gender, blood type and race/ethnicity). The owner of the record grants access to medical providers to add entries that pertain to different categories as drug allergies, a wellness category which includes blood pressure, weight, hours slept and minutes of exercise, problems and symptoms, medications, test results (imaging and lab), procedures and surgeries, and immunizations.

PHM has three types of users:
- Administrators
- Medical Providers
- Patients

The administrators can add new users, update their profile information, view users and activate/deactivate accounts and view an audit log of every transaction that happens in the system.

Patients can view and change their profile information and view their PHRs. They can also grant access tokens to a specific medical provider for a specific amount of time allowing that provider to update the patient’s health record. In a future version they will be able to submit change requests to medical providers when they find errors.

Medical providers can see a list of PHRs for which they have a valid access token and choose one of these PHRs to update. In a future version, they will be able to answer change requests submitted by patients.

Our PHM design uses a role-based model (on three levels) for granting access to information. First, navigation allows users to go only to pages to which they have access. Second, the application integrates a tool called Apache Shiro that prevents users from visiting any page (even if they type the URL in directly, bypassing intended page navigation flow) for which they do not have access. They are immediately re-directed to the login page. Third, both patients and medical providers can get to a PHR view page for a PHR they have access to. On a page like this that multiple types of users can navigate to, only controls valid for the user’s role are rendered.

The system also uses access tokens to control access to a user’s health record. The user grants an access token valid for a user-specified length of time to a particular medical provider.

3.3 Extensions to current knowledge

Standards for PHRs are not defined properly yet. In fact, the government is funding $60 million in grants through the Office of the National Coordinator for Health Information and Technology (the Strategic Health IT Advanced Research Projects or SHARP) [20]. The SHARP program focuses on four trouble areas that have slowed the adoption of electronic sharing of medical data: the first of these is security.

Of course, one of the big concerns is how the data is kept private, and another is how the data is kept accurately.

The accuracy of the data is governed by the ‘integrity’ security principle. In existing models, one party is responsible for the accuracy of the information. In the standalone PHR, as described in section 3.1, the patient owns her record and either does the updates herself (as in Google Health) or has a third party do them for her [12]. In the integrated and tethered models, the medical provider/health plan owns and maintains the data [12].

In our hybrid model for PHM, as opposed to an integrated or tethered PHR, the patient maintains control of the record. However, instead of having the user make the updates, the user grants an access token to a medical provider to make any changes. The access token is valid for a user-specified amount of time and is granted to a particular provider. The user reviews the record for accuracy. In a future version, the application will allow the user to submit change requests to the medical provider.

The benefits of this hybrid model can be listed as follows: the patient maintains ownership of his health record; no one may update a patient’s health record without being granted an access token by the patient; a medical provider’s access is automatically revoked when the access token expires; the patient’s health record is, presumably, more accurate because only medical professionals make updates (less likely to make mistakes with medical jargon and test results than common user); a patient retains responsibility for reviewing his health record and, in future versions, making change requests; the health record is portable across medical providers and insurance carriers; and the patient’s record maintains a state of integrity (valid, accurate data that can be counted...
upon in an emergency situation) through this system of checks and balances.

3.4 Uses for Implementation

Individuals wishing to be active participants in the management of their healthcare and the medical providers who care for them are the target audience for this application. The system provides a way for users to track all health-related information in one place and provides a system of checks and balances for maintaining the integrity of the data. Having this data congregated allows the user and individuals with whom she shares the record (e.g. health care providers) to view all data regarding health in one place and to feel reasonably sure that data is accurate.

There are many advantages to using a PHR [3] including better management of wellness care, minimization of medication interaction, reduction of medical test duplication, identification of health trends, and presentation of congregated data in logical format to healthcare providers. These advantages, however, could be easily be offset by allowing this private data to become available (for viewing or modification) to unauthorized users. This possibility makes security a critical component of the system [3].

4. Experimental work and results

Our PHM prototype is implemented with a combination of recent technologies as depicted with their inter-relationships in Figure 1. Use of current technology and selection of each component makes our design fast, unique, and adaptive for future extensions.

The first is a group of open source technologies commonly referred to as the LAMP (Linux/Apache/MySQL/PHP/Perl/Python) stack. The LAMP stack is frequently used for web application development because of the benefits it offers: access to source code of components, high developer productivity because components are relatively lightweight in terms of complexity, easy access to learning and reference material and a large talent pool [22]. The LAMP stack is less prevalent in large-scale corporate and government shops because it lacks vendor support (because the products are open-source) as compared to commercial offerings and because performance can be an issue with interpreted languages such as PHP.

An alternative to the open source stack is a technology stack based on Java Enterprise Edition (JEE). As seen in Figure 1, the following is the technology set we used to implement our PHM: JBoss 6 (Java application server), Java 6, MySQL 5, Java Enterprise Edition (specification for developing enterprise applications) which includes Java Server Faces 2.0 (Java web development framework), Java Persistence API (JPA), Hibernate implementation, Enterprise Java Beans 3 (EJB 3), and Apache Shiro (Java security framework).

This technology set offers a number of advantages. The portability of the Java Virtual Machine allows the application to be deployed on anything from a handheld device to an IBM Z Series mainframe. These applications have broad industry support and are heavily used for large-scale corporate and government system development. Java offers the performance benefits of a compiled language. Java provides the JAVA 2 security model that provides high-level control over access to system resources. Like the open source stack, a large talent pool is available. Finally, Java is ubiquitous; almost every product has a Java API.

The work in [23] compares common vulnerability density (CVD) of open source web applications written in Java and PHP and concludes that Java applications have a “substantially lower CVD than similar applications written in PHP.” The same study, however, also concludes that programming language is not an important consideration while developing web applications since the variation in CVD is much greater between projects than between languages. This conclusion indicates that programmer capability is more important than language choice.

Our choice of technology in designing PHM is based on the idea that a large-scale implementation of a PHR application needs an enterprise-level technology set.
Figure 2 depicts the detailed overall page flow diagram in a Finite State Machine (FSM) format for our PHM design.

As seen in Figure 2, our PHM scheme provides an easy to navigate platform with a detailed and complete consideration of different states and transitions.

4.2 Security concerns

Any time a person’s health information is aggregated in one place, particularly electronically, there are significant security concerns. We consider the ITU-T X.800 security architecture [24] to incorporate the fundamental principles of security: confidentiality, integrity, availability, authentication, authorization and non-repudiation in our PHM. As mentioned in the X.800 recommendation [25], these security services can be implemented in various layers of the Open Systems Interconnection (OSI) model: from physical layer through application layer. In our design, we adopt the application layer application of the well-known X.800 architecture, since the PHM prototype we propose operates at the application layer of the OSI model. This consideration strengthens the effect of our paper in the sense that it is a PHR scheme that implements a comprehensive consideration of security requirements, other than partially meeting them.

In some cases, we integrated software or made use of protocols to ensure the security requirements were met. The maintenance of the integrity of the data, however, required changes to the existing model.

In following subsections, we briefly define and customize the components of X.800 security architecture. It is important to embrace each security feature because our design involves all three probable states of data: in process, in transmission, and in storage.

4.2.1 Confidentiality. Confidentiality refers to restricting access to information to those who need to have access to it [25]. Two types of confidentiality breaches could occur. First, web traffic between the patient and the application on the server is susceptible to attack. To prevent access to private information, Secure Socket Layer (SSL) is employed to encrypt all web traffic. Second, unauthorized users could gain access to the database. Key fields (such as password and social security number) in the database are either encrypted or hashed to prevent connecting an unauthorized patient with his medical records.
4.2.2 Integrity. The reliability and accuracy of the data held in the personal health record are of vital importance. If the patient cannot be sure enough of the integrity of his data to count on it, or if a medical provider cannot be certain enough about the accuracy to act on it, the personal health record becomes useless. This makes the integrity of the data of the utmost importance.

Consideration of the security principle ‘integrity’ [25] always includes protecting against modification of the data and preventing replay, or the re-sending of a message. Both of these possibilities are eliminated with the use of SSL. SSL prevents replay by providing a timestamp in the Message Authentication Code (MAC) so the message will not be accepted more than once by the server and by providing a connection ID that is large enough to make ‘guessing’ the key, and then being able to masquerade as an authorized user, sufficiently difficult. To protect SSL against resource loading and SSL renegotiation attacks that could occur by client (patient’s host) overloading the server (PHM host) with more requests than it can handle, countermeasures such as iRule [26] can be taken.

In this particular application, however, the integrity of the data seems more likely to be threatened by inaccuracies in data entry. In the PHR model where the patient owns the record, mistakes entering medical jargon would be easy to make. In the EHR model where the insurance company or medical provider owns the data, data entry errors seem to be less likely as the owner of the record is familiar with medical procedures, however the owner may not be familiar with the patient or his history and may not be as adept at finding errors unrelated to the current entry.

The hybrid model we propose in this paper attempts to improve the integrity of data in the personal health record by solving these problems. The patient owns the record and maintains the responsibility of verifying the accuracy of the data and submitting change requests as needed to correct errors. The only people who can update the record, however, are medical providers with a valid access token granted by the patient.

This system of checks and balances attempts to capitalize on the strengths of each of the participants in the PHR system in order to provide the highest level of data integrity.

4.2.3 Availability. In addition to ensuring the availability of the data in the personal health record by allowing access to authorized users using the secure methods described in section 4.2.5, Denial of Service (DoS) attacks are prevented by the use of a timestamp in the SSL MAC. DOS attacks occur when a malevolent user resends the same message thousands of times, tying up the server and preventing valid users from accessing data.

4.2.4 Authentication. Authentication, or ensuring a person is who he claims to be [25], is another vital security principle, and much attention has been paid to ensuring proper authentication.

Of course, the password is masked to prevent the hijacking of an authorized login by stealing the password. The account is disabled after three incorrect login attempts to make brute-force attempts by a computer trying to guess a valid login/password combination.

Apache Shiro, a software security package, was integrated to provide both authentication and session management. Shiro provides authentication of the login/password combination. Shiro also validates that an authenticated user is currently logged in each time he navigates to a new page. If he is not, he is redirected to the login page. Additionally, a user is automatically redirected to an error page if he navigates to a page for which he does not have access.

Two-factor authentication is planned for the next version of this application. Two-factor authentication involves using two of three proofs to identify yourself [25], such as something you know (e.g. a password), something you have (e.g. a card with a grid on it; at each login, the user is prompted to enter the value from a different cell), or something unique about yourself that you are or you produce (e.g. iris, a fingerprint).

4.2.5 Authorization. After a user has been authenticated, the application still needs to be sure she has the appropriate authorization to access [25] a PHR. Authorization is checked using two vehicles: access tokens and roles.

A patient grants access tokens to medical providers that are valid for a patient-designated amount of time. The access token is implemented as a record in the database containing the patient’s and medical provider’s identifications and a timestamp indicating when the token expires. When a provider navigates to a page, a database query returns only PHRs to which the provider has a current token. (The tokens themselves never leave the server and are protected by the authentication considerations in section 4.2.4 and by role-based authorization described later in this section.) Without a valid, unexpired token, a provider does not have access to that patient’s health record. A future improvement to the access token idea is to provide different levels of
A role-based approach to authorization is implemented on three separate levels: First, each user is assigned a role (patient, medical provider or administrator). The user can navigate only to pages for which her role is authorized. Second, Apache Shiro is integrated with this application. URL patterns are set up in Shiro based on role. If a user navigates to a page to which she does not have access (based on her role), she is redirected to an error page. For example, Alice is logged in with the role of patient. She somehow learns the URL for the administrator-only page that would allow her to add users and she enters that URL directly into the browser’s address bar. Before the application is directed to navigate to that page, Shiro checks her role against the accepted role for that page. Because she is not authorized to view that page, she will be redirected to an error page. Third, if users with different roles can access a page, only role-appropriate controls on that page will be rendered. This is accomplished by writing custom tags for Shiro. For example, there is an application page where both medical providers and patients can view a PHR. If a patient navigates to that page, his own PHR is displayed along with a button allowing him to navigate to a page where he can grant access tokens to medical providers. If a medical provider navigates to that page, he is shown the PHR for the patient whose PHR he is updating. Unlike the patient view, the fields are updatable in the medical provider view and the medical provider may not grant access to anyone else.

4.2.6 Non-repudiation. Non-repudiation means preventing the denial (by any party using the application) that an action occurred [25]. A report on the viability of use of PHRs from the National Committee on Vital and Health Statistics [26] indicates that a user should be able to audit when and by whom his PHR has been accessed.

An audit log is used to satisfy both the non-repudiation security principle and the recommendation from NCVHS.

Every database event in the system is logged with a timestamp and the IP address of the logged in user. In addition, all database manipulations occur in the same transaction with the corresponding audit entry. If one or the other does not occur, the transaction is rolled back, ensuring that the audit log and database access are in sync.

A database administrator with malicious intent could still bypass the security shield of our PHM. So, a future improvement is to add a database trigger so that the audit entry is done by the database.

In Table 2, we adopt the security services suggested by the X.800 architecture in [25], and summarize the security components of our PHM scheme that help shield it against various security attacks.

<table>
<thead>
<tr>
<th>Table 2. Security shield of our PHM scheme.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Confidentiality</strong></td>
</tr>
<tr>
<td>● Encrypt key fields in database</td>
</tr>
<tr>
<td>● Use SSL to encrypt message between</td>
</tr>
<tr>
<td>web server and browser</td>
</tr>
<tr>
<td><strong>Integrity</strong></td>
</tr>
<tr>
<td>● Restrict PHR updates to medical</td>
</tr>
<tr>
<td>professional</td>
</tr>
<tr>
<td>● Use SSL which provides timestamp</td>
</tr>
<tr>
<td>in MAC and large connection ID</td>
</tr>
<tr>
<td><strong>Availability</strong></td>
</tr>
<tr>
<td>● Use SSL (MAC contains sequence number</td>
</tr>
<tr>
<td>to prevent replay attacks)</td>
</tr>
<tr>
<td><strong>Authentication</strong></td>
</tr>
<tr>
<td>● Use Apache Shiro to authenticate</td>
</tr>
<tr>
<td>login and manage sessions</td>
</tr>
<tr>
<td>● Employ account disabling</td>
</tr>
<tr>
<td>● Two-factor authentication for medical</td>
</tr>
<tr>
<td>providers</td>
</tr>
<tr>
<td><strong>Authorization</strong></td>
</tr>
<tr>
<td>● Limit access to data (through program</td>
</tr>
<tr>
<td>design and Apache Shiro) to users with</td>
</tr>
<tr>
<td>appropriate role</td>
</tr>
<tr>
<td>● Limit access to PHR (through access</td>
</tr>
<tr>
<td>token system) to patient’s chosen</td>
</tr>
<tr>
<td>medical providers</td>
</tr>
<tr>
<td><strong>Non-repudiation</strong></td>
</tr>
<tr>
<td>● Every database event is logged in the</td>
</tr>
<tr>
<td>audit log</td>
</tr>
<tr>
<td>● Database access and audit updates are</td>
</tr>
<tr>
<td>managed in the same database</td>
</tr>
</tbody>
</table>

4.3 Comparison

To provide a systematic comparison of our PHM, we selected one example PHR representative of each model as follows: Microsoft HealthVault and Google Health for standalone PHR, Health Record Bank (HRB) for integrated PHR, Aetna for tethered PHR, and our PHM for hybrid PHR.

The example products will be compared based on security, ownership of data, whether they contain data from multiple sources, how updates are made, whether data is portable across medical or insurance providers and whether they include a review process.

The standalone PHR example, Microsoft’s HealthVault makes no mention of security on its website. The patient owns the data, and the PHR can contain data from many sources including pharmacies, user-generated data and medical providers. The user inputs much of the data (such as immunization records), but some data can be
imported from health and fitness devices or uploaded from certain providers that work with Microsoft (such as CVS pharmacy) [15]. There is not a built-in review process.

An implementation of an integrated PHR that is gaining in popularity is Health Record Banking (HRB) [12]. The Health Record Banking Alliance [31] is an organization of technology professionals that is devoted to promoting the creation of HRBs. According to [31], which is talking about the concept of HRBs, not a specific product, data would be stored securely using the strongest data encryption methods available. HRBs provide three fundamental transactions: deposit of new medical records, withdrawal of those records by healthcare providers and search of records. Ownership of data would remain with the HRB [12]. Data from many sources could be maintained.

Aetna’s PHR implementation is a good example of a tethered PHR [16]. Its website states that it is password protected and governed by state and federal privacy laws. Aetna will only share information with doctors approved by the patient and, although it will not use a patient’s information for underwriting purposes, it does have access it. Aetna maintains ownership of the information, and a patient can only be a user of the site as long as he is an Aetna customer. The data is exportable to Microsoft’s HealthVault in case a patient changes insurance plans. Aetna’s PHR will maintain data from multiple providers and can be updated automatically by some pharmacies, doctors and labs.

Although detailed security information is not available from most PHR providers, they presumably take security into consideration. However, by definition they do not provide the enhancement to the integrity security principle that our model provides. In summary the primary differences are first, that the patient maintains ownership of the data and the medical provider, specifically authorized by the patient, updates it. Second, PHM contains an integrated review process by which the patient can submit change requests and the medical professional can answer them. The combination of these improvements to the model helps ensure a higher level of integrity of the patient’s data. Having a higher level of integrity improves security and will help further the use of PHRs.

5. Conclusion and future work

We present the design and development of a hybrid personal health record system equipped with comprehensive security considerations. Security in electronic health record keeping will be an important issue in the next decade, and our PHM system proposes to be a step toward exploring concerns involved. Our system explicitly introduces the additional comprehensive security shield that comprises the security parameters of the ITU-T X.800 architecture to PHR processing. We compare our design with its peers that belong to each of the standalone, integrated, tethered, and hybrid models based on different parameters as runtime, security, technology cost, and ease of use.

Our PHM system is a base model to begin exploring security concerns in a web-based personal health record system. There are a multitude of extensions to this project, many of which will introduce new security concerns (particularly the extensions which involve new technologies such as smart cards).

There are many ways to extend this work, but the most important is to add code that allows users to make change requests (request a medical provider to make a change/correct an error) in the PHR and to allow medical providers to answer those requests. This piece is necessary to complete the model envisioned where the user owns the record, but trained medical providers make all changes allowing a higher degree of integrity for the data.

Extending the access token idea to grant different levels of access to different providers (perhaps give less access to a dentist or pharmacist than an internist) would add a new level of control for the user.

Adding a ‘break the glass’ feature, a way for emergency personnel to access the record under very specific conditions will help the application meet standards set forth by the National Committee on Vital and Health Statistics [29].

As future work, we also plan on extending our current PHM design to include an audit entry feature that is accomplished through database so as to eliminate any malevolent administrator activities.

A PHR that we describe and implement in this work can be used as a basis for research to explore means to implement its security features in different layers of the OSI architecture. In the hope to improve their efficiency, application of each security measure can be performed in different layers to obtain maximum outcome.

Consideration of the security of data at rest, i.e. the data that resides on peer hosts is handled through use of various security measures such as encryption, employing hash functions, and SSL in the development of our prototype. Further exploration of such measures to develop a fully functional scheme
that complies with the HIPAA requirements is another avenue that the authors plan on to follow.

6. References


[31] Department of Health and Human Services, Office of the National Coordinator for Health Information Technology, Defining Key Health Information Technology Terms, report to the Office of the National Coordinator for Health Information Technology, 2008.