Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/cbm

An open platform for personal health record apps with platform-level privacy protection

P. Van Gorp [a],*, M. Comuzzi [b], A. Jahnen [d], U. Kaymak [a], B. Middleton [c]

[a] *Eindhoven University of Technology, The Netherlands*
[b] *City University London, United Kingdom*
[c] *Partners HealthCare and Harvard Medical School, MA, USA*
[d] *Public Research Center Henri Tudor, Luxembourg*

ARTICLE INFO

Article history:
Received 7 February 2013
Accepted 28 April 2014

Keywords:
Personal health records
Apps
Architecture
Trust
Privacy

ABSTRACT

One of the main barriers to the adoption of Personal Health Records (PHR) systems is their closed nature. It has been argued in the literature that this barrier can be overcome by introducing an open market of substitutable PHR apps. The requirements introduced by such an open market on the underlying platform have also been derived. In this paper, we argue that MyPHRMachines, a cloud-based PHR platform recently developed by the authors, satisfies these requirements better than its alternatives. The MyPHRMachines platform leverages Virtual Machines as flexible and secure execution sandboxes for health apps. MyPHRMachines does not prevent pushing hospital- or patient-generated data to one of its instances, nor does it prevent patients from sharing data with their trusted caregivers. External software developers have minimal barriers to contribute innovative apps to the platform, since apps are only required to avoid pushing patient data outside a MyPHRMachines cloud. We demonstrate the potential of MyPHRMachines by presenting two externally contributed apps. Both apps provide functionality going beyond the state-of-the-art in their application domain, while they did not require any specific MyPHRMachines platform extension.

1. Introduction

Without the participation of the patient, a health care provider cannot effectively treat (or prevent) disease-causing behaviors. The doctor–patient relationship is therefore gradually evolving from a paternalistic approach to a more participatory model [1,2]. Houston and Ehrenberger argue that a key factor for successful patient participation is information sharing: patients require good information not only to care for themselves, but also to effectively communicate with their physicians [3]. Empowering the patient with information is particularly important since information exchange between different caregivers is very limited [4], especially beyond the scope of local business networks (such as the Partners HealthCare system in the US state of Massachusetts or the The Eye Care Network in the Netherlands [5,6]).

The two key stakeholders in this scenario, i.e., patients and their physicians, are often willing and capable to share information. Already before the turn of the millennium, for instance, various online surveys demonstrated high adoption rates of e-mail as a patient-provider communication medium [7]. E-mail information sharing, unfortunately, has several limitations. Most notably, message exchanges are completely ad hoc, preventing patients to build and maintain a longitudinal record of their health data, to use the integrated record to effectively care for themselves, and to share all their health data effectively and securely with their caregivers.

To overcome these limitations, Personal Health Record (PHR) systems have been proposed by various companies and authors in academia [8]. PHR have many societal benefits, such as empowering patients in the management of their own health and fostering interoperability among health care providers, possibly reducing the overall costs of diagnosis and treatment [9]. Policy makers, therefore, have repeatedly called for technologies that “enable patients, doctors and other health care providers to access personal health records securely through the Internet, no matter where a patient is seeking medical care” [10,11]. Unfortunately, PHR adoption levels in practice are very low due to privacy concerns as well as the lack of convincing medical and business use cases. The US department of Health and Human Services, for instance, has invested heavily with the expectation that “once the market has structure, patients, providers, medical professionals and vendors will innovate, create efficiencies and improve care” [10].

One of the reasons for the low level of adoption of PHRs is their lack of openness at the platform level. Mandl and Kohane [12] have addressed the issue by looking at positive and negative

* Corresponding author.
E-mail address: p.m.e.v.gorp@tue.nl (P. Van Gorp).

http://dx.doi.org/10.1016/j.compbiomed.2014.04.019
0010-4825/© 2014 Published by Elsevier Ltd.

experiences from various health record projects. The authors conclude that PHR technologies should go beyond the “conventional” requirements for Electronic Health Record (EHR) technologies, i.e., interoperability, security, and privacy. PHR systems should support open innovation and, therefore, they should (a) reduce impediments to the transfer of data, (b) provide substitutable software components, i.e. “apps”, and (c) they should allow competition and “natural selection” for high-value, low-cost software components. Regarding substitutability, the authors clarify that PHRs should enable the combination of software components developed by different vendors without creating impediments to replace such components over time [12].

In this paper, we propose the use of MyPHRMachines, a PHR platform that satisfies the above requirements. The platform is unique in its openness: it presents the least possible impediments to the transfer of data and it *prevents* apps from violating privacy requirements by design. These properties are based on the use of Virtual Machines (VMs) as flexible and secure execution sandboxes for the apps. To show the effectiveness of the approach, we discuss externally contributed apps for Radiation Exposure Measure (REM). As we will show later, radiology and, more specifically, REM, is a typical application scenario that can benefit from an open PHR platform.

The remainder of this paper is structured as follows. Section 2 discusses the shortcomings of current PHR platforms with regards to openness. Section 3 describes the MyPHRMachines platform. Section 4 gives the motivation for and presents the REM application scenario, while Section 5 describes the REM apps in MyPHRMachines. Finally, Section 6 discusses the contribution of the paper by providing a link also to the PHR literature.

2. Openness of PHR platforms

Opening a platform enables its owners to strategically disclose aspects related to the development or commercialization of the platform [13].

There are broadly two different approaches to opening a platform. The first entails giving up some *control* over the platform, whereas the second entails only granting *access* to the platform to outsiders [14]. When a company devolves all *control* over a platform, there is no longer a single party who controls its evolution. In terms of PHR platforms, this would mean for example that the development activities for a platform are opened up to the open source community, or to selected commercial software vendors. The Indivo platform is the primary example of this form of PHR platform openness [15]: starting from a development project at the Harvard Medical School and Massachusetts Institute of Technology, the project was then opened to the open source community as well as to Google, Microsoft and other commercial partners.

The second form of openness (granting *access*) implies that the platform owner maintains control over its core development while relying on the market to provide complementary innovation around it. Apple's App store is a well known example of this approach, where the company not only preserves control over the platform's development, but even controls the transactions on the platform. Microsoft HealthVault is a well known PHR platform that is open to apps from third party developers, while Microsoft controls the core platform [16].

We position the novelty of our PHR platform in the latter category. MyPHRMachines provides *app developers* with open *access* to the app platform, but it guarantees that patients can trust the platform in the protection of their personal data.

As illustrated in the remainder, other PHR platforms are either (1) completely closed or (2) pose too tight restrictions on the type of data that can be managed by the platform. In the latter case, technical guarantees regarding the *prevention* of data abuse are completely missing. Therefore, for those PHR platforms that grant app developers access to deploy their apps, access is only granted to trusted parties that can be held liable in case they violate their promises to the platform provider and end users. MyPHRMachines makes such app-specific trust considerations irrelevant, since technical privacy protection measures are already implemented at the platform level. Consequently, a MyPHRMachines-based App store can be opened up securely also to non-trusted app developers.

PHR system architectures can be classified into provider-tethered and free-standing ones [17]. For the provider-tethered variant, the PHR system is essentially a portal extension of Hospital Information Systems (HISs), which only contain data from one health care provider or institution. Examples in this category are EPIC MyChart [18] and MyHealtheVet [19], tethered from the EPIC EHR and the HIS of the US Department of Veterans Affairs, respectively. Free standing PHRs are stand-alone PHR platforms, which can store data generated and provided by various health care institutions or by the patient. Examples in this category are HealthVault and Indivo version *X* [15]. In principle, this classification only considers the stakeholder controlling the PHR platform (a single health organization versus an independent party). In practice, all tethered PHR systems are completely closed, while some free-standing PHR systems make their platform accessible to external app builders. Still, there are fundamental issues even for free-standing solutions. Below, we discuss some of these issues for the cases of Microsoft HealthVault and Indivo X.

Microsoft HealthVault provides a set of libraries (e.g. for Java and .NET developers) to Create, Read, Update, and Delete (CRUD) all types of data in the HealthVault system. The libraries are based on a Web service API. Similarly, Indivo X enables external software to perform CRUD operations on its health data through XML-based standard data models. Indivo X is also integrated with SMART [20], a more general solution to support the exchange of health data among health institutions. SMART provides an OWL-DL ontology to semantically annotate health data. Unfortunately, for both platforms, two of the requirements elicited in Section 1 are not satisfied:

1. existing platforms do not *actively prevent* apps from violating end-user privacy requirements, and
2. existing platforms pose*impediments on the transfer of health data*.

The first issue relates to Mandl et al.'s *conventional* requirements for EHR systems, while the second one relates to their extra requirements for openness.

The privacy issue is caused by the fact that neither HealthVault nor Indivo X apps are executed inside a controlled ecosystem. Instead, app code is executed on a third party infrastructure and, if users grant an app access to load PHR data, then that data can travel freely to the servers of the app providers. In terms of liability, the platform providers (Microsoft and others) push responsibilities to the app builders and the end-users. This implies that (i) all app builders need to provide terms of use agreement that promises that patient data will not be abused and (ii) end-users need to review and consent such agreements for each and every app. While such agreements can protect end-users ex-post (e.g., legally) they do not physically prevent app providers to maliciously use the PHR data behind the scenes. Also, for app builders not interested in patient data, this need for app-specific data use agreements forms an undesirable barrier to entering the app market.

The second issue, i.e., impediments on the transfer of health data, is caused by the fact that data can only be stored on the