A Context-Aware, Capability-Based, Role-Centric Access Control Model for IoMT

Flora Malamateniou(✉), Marinos Themistocleous, Andriana Prentza,
Despina Papakonstantinou, and George Vassilacopoulos

Department of Digital Systems, University of Piraeus, 80 Karaoli and Dimitriou street,
Piraeus, Greece
{flora,mthemist,aprentza,dpap,gvass}@unipi.gr

Abstract. The Internet of Medical Things (IoMT) can be described as connecting everyday devices and wearables to the Internet in order to intelligently link them together, thus enabling new forms of communication between things (medical devices) and people (patients) and between things themselves. This paper describes a context-aware access control model that hinges on the role-based and attribute-based access control (RABAC) and the capability-based access control (CapBAC) models. A prototype access control mechanism based on the model is intended to be incorporated into a personal health record (PHR) platform.

Keywords: IoMT · Access control · Role-centric model · Capabilities-based model

1 Introduction

Recently, there has been a noted shift in the healthcare sector from episodic-care to continuous-care and from traditional hospital care to connected care when and where possible [2, 9]. As a result, there is growing interest for home care services in order to support better health management and independent living, especially for the elderly [1, 9, 10].

Basically, home care services aim at providing continuous patient monitoring and immediate response by healthcare professionals in case of either emergency situations, indicated by abnormal physiological data and medical measurements, or risk situations, indicated by patient abnormal behavior (e.g. less movement, lack of personal care). Moreover, it has been suggested that continuous, connected healthcare, based on the "Internet of Medical Things (IoMT)" is an important component of smart cities of the present and of the future, since, increasingly, patients want to heal at home and connected care at home allows this to happen [1, 9, 10]. Hence, connected healthcare is an alternative to institution-based healthcare that has not been designed to keep up with demand or the desire for patients to heal and age at home.

Along with growth, healthcare organizations have been developing a whole new digital health strategy. Instead of using systems that push information from one hospital and physician to the other, they employ personal health records (PHRSs) and patient portal technology that encourage patient engagement and patient-centered care [2–4]. For the

P. Perego et al. (Eds.): MobiHealth 2016, LNICST 192, pp. 125–131, 2017.
DOI: 10.1007/978-3-319-58877-3_16

patient, the PHR stores all of the patient data in one place and is easy to use providing access to patient data and more online, round the clock and at their convenience, which is fulfilling their needs and demands. However, when IoMT are integrated around a PHR platform, various security challenges have to be confronted, such as patient privacy, end-to-end security, user authentication, access control and resilience to attacks [3, 6].

This paper presents a context-aware capability role and attribute-based access control (CapRABAC) model that has been developed for connected medical devices that have been integrated around a PHR platform and hinges on the role and attribute-based access control (RABAC) and the capability-based access control (CapBAC) models [3, 5–8]. Thus, through a capability transfer the model allows to delegate to certain role holders both contextually constrained permissions and roles.

2 Motivation

The basic motivation of this research stems from our involvement in a recent project concerned with development of an authorization mechanism for connected medical things. Table 1 shows an extract of authorization policies regarding PHR and medical device-produced data accesses by physicians.

Table 1. Extract of **authorization** policies

No	Authorization policy
1	A physician assigned to treat a patient (patient's attending physician) is allowed to access the PHR and to read data produced by connected medical devices that have been assigned to the patient and are related to his/her specialty
2	A physician assigned to do a night or holiday duty (duty physician) is allowed to access the PHR and to read data produced by connected medical devices that have been assigned to the patients and are related to his/her specialty
3	A physician requested to provide consultation on a patient (patient's consulting physician) is allowed to access the PHR and to read data produced by connected medical devices that have been assigned to the patient and are related to his/her specialty, on delegation by the patient's attending physician

The authorization policies of Table 1 surface certain permission delegation and propagation requirements with regard to the three physician (functional) roles involved: attending physician, duty physician and consulting physician. In fact, the role "attending physician" corresponds to a relationship (physician, patient), the role "duty physician" corresponds to a relationship (physician, healthcare unit) and the role "consulting physician" corresponds to a relationship (attending physician, physician). These roles are revoked when the relationship occurrences cease to exist.

Since PHRs empower patients with data ownership, it is the patients themselves who are entitled to grant permissions on accessing their own data to others. However, this may be proved infeasible in the stressful medical environment (e.g. in emergency cases or, even, in ordinary cases). Hence, there is a need to somehow automatically allow physicians receive the least possible authorizations (just-in-time and at the suitable level of granularity) for performing their healthcare activities.

A realistic scenario of the need for automatic delegation of authority (the data access permissions that have been assigned to the role) is as follows: Upon assignment to a patient, a physician takes up the role of the "attending physician" for the patient automatically; upon request to provide consultation for a patient, the requested physician takes up the role of the "consulting physician" for the patient automatically; and, upon assignment to a night or holiday duty, a physician takes up the role of the "duty physician" automatically.

3 Capability-Based RABAC Model Overview

On the above premises, a context-aware capability role and attribute-based access control (CapRABAC) model has been developed for controlling access to data objects from connected IoMTs which are integrated around a PHR platform [4, 11]. Basically, CapRABAC is an extension of the RABAC model obtained by integrating a CapBAC mechanism into a context-aware RABAC model (that dynamically adjusts role and permission assignments based on contextual information so that users are provided with tight, just-in-time permissions). Moreover, the model addresses the issue of flexible delegation of capabilities subject to capability propagation constraints [5, 6, 8]. To reduce the complexity of the model only positive authorizations are considered.

Figure 1 shows the context-aware CapRABAC model that includes the following main components:

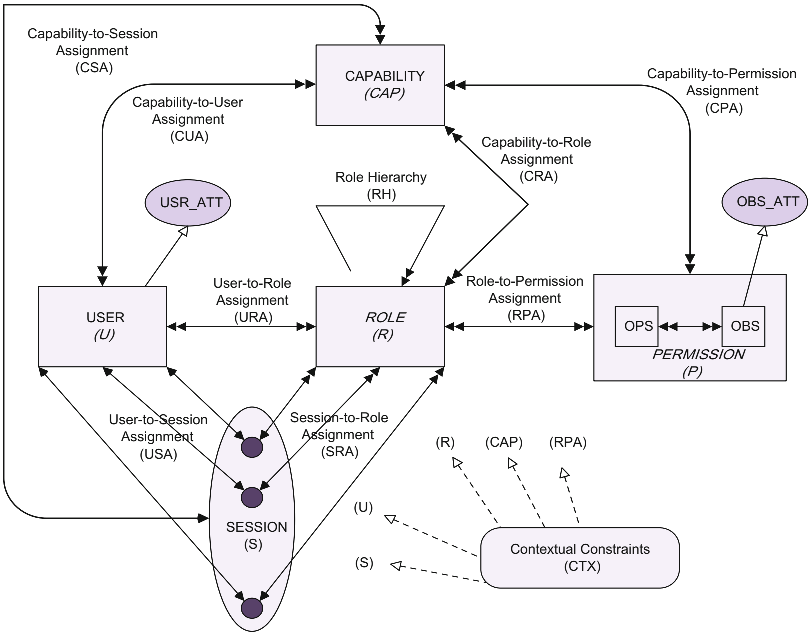
Capability-to-Session Assignment (CSA)
CAPABILITY (CAP)
Capability-to-Permission Assignment (CPA)
Capability-to-User Assignment (CUA)
Capability-to-Role Assignment (CRA)
Role Hierarchy (RH)
USR_ATT
OBS_ATT
USER (U)
User-to-Role Assignment (URA)
ROLE (R)
Role-to-Permission Assignment (RPA)
OPS
OBS
PERMISSION (P)
User-to-Session Assignment (USA)
Session-to-Role Assignment (SRA)
SESSION (S)
(R)
(CAP)
(RPA)
(U)
(S)
Contextual Constraints (CTX)

Fig. 1. The context-aware CapRABAC model.

- *USER:* A user is an entity whose access is being controlled. Patients are super-users as the owners of their PHR data.
- *ROLE:* A role may be assumed as a named collection of capabilities or permissions which define the division of work and the lines of authority based on job functions and seniority. Thus, roles encapsulate sets of permissions, they are assigned to users through a many-to-many relationship and they may be activated at run time, possibly with regard to contextual constraints. Thus, roles encapsulate the minimum sets of permissions which are tailored to run time access needs of users by taking into account the context (i.e. are constrained by context rules) so that users are provided with tight, just-in-time permissions.
- *PERMISSION:* A permission is an approval to access one or more protected objects in certain modes.
- *CAPABILITY:* A capability is a communicable, unforgeable token consisting of an object identifier and a list of permitted operations for that object. Hence, a capability represents a self-authenticating permission to access a specific object in permitted operations, whereby owners of the capability can access the object without any authentication. Moreover, it is used for delegation of authority, usually achieved in three steps performed by the delegator (i.e. a user who wishes to delegate authority to another user): (1) capability creation, (2) assignment of some permissions and/or roles to the capability, and (3) transfer of the capability to the intended delegate (i.e. receiver of the capability). Thus, a user can delegate both permissions and roles by a capability transfer.
- *SESSION:* A session is a set of user interactions with the system during which a user is assigned a set of roles. One of these roles will be active for each interaction. Role activation and deactivation may be performed dynamically during a session.
- RH $\subseteq$ ROLE $\times$ ROLE, a partial order on all roles, called the role dominance relationship (also, indicating which functional roles can be derived from which organizational roles).
- URA $\subseteq$ USER $\times$ ROLE, a many-to-many user to role assignment relationship (indicating the roles assigned to users).
- RPA $\subseteq$ ROLE $\times$ PERMISSION is a many-to-many role to permission assignment relationship (indicating the permissions assigned to roles).
- USA $\subseteq$ USER $\times$ SESSION is a one-to-many user to session relationship (indicating the sessions activated by a user).
- SRA $\subseteq$ SESSION $\times$ ROLE is a many-to-many role to session relationship (indication the roles activated during a session).
- CUA $\subseteq$ CAPABILITY $\times$ USER is a many-to-many capability to user to assignment relationship (indicating both the delegators and the delegates of capabilities).
- CRA $\subseteq$ CAPABILITY $\times$ ROLE is a many-to-many capability to role assignment relationship (indicating the roles that are delegated by capabilities).
- CPA $\subseteq$ CAPABILITY $\times$ PERMISSION is a many-to-many capability to permission assignment relationship (indicating the permissions that are delegated by capabilities).
- CSA $\subseteq$ CAPABILITY $\times$ SESSION is a many-to-many capability to session relationship (indicating the capabilities that are activated during a session).

The model of Fig. 1 indicates that some additional components have been introduced to the original RABAC model, namely, a set of capabilities, a mapping to determine the owners of the capabilities and the assignment of roles and permissions to capabilities. Note that in terms of these assignments both roles and permissions are treated as delegation units.

3.1 Contextual Rules

Context is a set of context types evaluated during a session (at run time) which constrains the available set of access permissions. Context types may correspond to domain-dependent or domain-independent concepts. For example, with regard to medical device data accesses, domain-dependent context types may be user (userCtx) and object (objCtx) and domain-independent context types may be time (timeCtx) and location (locCtx).

Based on the context types, three kinds of context components may be defined: (a) contextual attributes (corresponding to user and object attributes of the RABAC model), (b) contextual sets (corresponding to set valued attributes of the RABAC model), and (c) contextual functions (corresponding to the filter functions that are boolean expressions based on user and object attributes of the RABAC model) [7]. These are expressed in the syntax <context type>.<component>.<name>.

For example, in a medical situation, the context type *userCtx* may include the contextual attribute *(userCtx.Att.user_id)*, representing user identity, and may be related to the contextual set (userCtx.Set.physician), representing that the user is actually a physician, and to the contextual function (userCtx.Fn.attending(pat_id)), specifying that the user is an attending physician to the patient with the particular pat-id. Hence, a role defined by a relationship between a physician and a patient (e.g. attending physician, consulting physician) may be represented by a contextual function.

Contextual rules relate contextual information, considered relevant to a particular situation, in logical expressions that constrain an access control policy with regard to protected objects. Essentially, they are parameterized expressions whose arguments are evaluated at run time to determine whether an attempted access should be permitted (if they evaluate to *"True"*) or denied (if they evaluate to *"False"*). For example, the rule *homecare(pat-id) {pat-id in pat.Ctx.Set.home_care_patients}*, where "*in*" checks whether *pat-id* belongs to the set of home_care_patients of a hospital or a health district, yields *"True"* if the patients identified by *pat-id* is a homecare patient.

3.2 Authorization Rules and Capabilities

On the above considerations, the authorization rules *AUTH* and the capabilities *CAP* of the context-aware CapRABAC model for IoMT are of the form

$$
\begin{aligned}
&AUTH = \{R, (M,\ O/P/MD), CON, \delta\}\ \ /Authorization\,rule\\
&CAP = \{R, (M, O/P/MD), CON, \delta\}\qquad /Permission\,capability
\end{aligned}
$$

The authorization rule *AUTH* indicates that a user holding the role R may exert permission (M, O/P/MD), (for performing operation M on the object O/P/MD of medical device MD connected to patient P), under a set of contextual constraints (expressed as contextual rules) CON that must be fulfilled (evaluated to *"True"*) and that this permission may or may not be passed on by the role holder as denoted by flag δ. Similar interpretations hold for the corresponding capability *CAP*.

This definition presumes that users holding specific roles may be granted sets of permissions by a role holder who holds all the permissions (in the case of a PHR platform, the latter user holds the *"patient"* role who owns his/her data). Subsets of these permissions may then be passed on by the grantees to other role holders.

4 Concluding Remarks

Success of connected care largely depends on protecting patient information security and privacy. This paper presents CapRABAC, an access control model for healthcare information stored into a PHR and connected medical devices. The model is an attempt to provide a flexible approach for managing information security with regard to medical devices used by patients and connected to a PHR platform. Hence, the use of ontologies has been adopted that results into suitable context-aware authorizations for users. Model implementation into a mechanism, which is intended to be incorporated into an experimental PHR platform, is still in progress focusing, among the rest, on intelligent, ontology-based approaches for automatic delegation of roles and of context-aware permissions and for enabling such devices, agents and services interact securely and with the least user intervention.

References

1. Bhide, V.: A survey on the smart homes using Internet of Things (IoT). Int. J. Adv. Res. Comput. Manage. **2**(12), 243–246 (2014)
2. Calvillo, J., Roman, I., Roa, L.M.: Empowering citizens with authorization mechanisms to their personal health resources. Int. J. Med. Inform. **82**, 58–72 (2013)
3. Carrion, I., Aleman, J., Toval, A.: Accessing the HIPAA standard in practice: PHR privacy policies. In: Proceedings of the 33rd Annual International Conference of the IEEE EMBS, Boston, Massachusetts, USA (2011)
4. Chen, T.S., Liu, C.H., Chen, T.L., Chen, C.S., Bau, J.G., Lin, T.C.: Secure dynamic authorization scheme of PHR in cloud computing. J. Med. Syst. **36**(6), 4005–4020 (2012)
5. Gusmeroli, S., Piccione, S., Rotondi, D.: A capability-based security approach to manage access control in the Internet of Things. Math. Comput. Model. **58**(5–6), 1189–1205 (2013)
6. Hernandez Ramos, J., Jara, A., Marın, L., Skarmeta, A.: Distributed capability-based access control for the Internet of Things. J. Internet Serv. Inf. Secur. (JISIS) **3**(3/4), 1–16 (2013)
7. Jin, X., Sandhu, R., Krishnan, R.: RABAC: role-centric attribute-based access control. In: Kotenko, I., Skormin, V. (eds.) MMM-ACNS 2012. LNCS, vol. 7531, pp. 84–96. Springer, Heidelberg (2012). doi:10.1007/978-3-642-33704-8_8
8. Li, F., Rahulamathavan, Y., Conti, M., Rajarajan, M.: LSD-ABAC: lightweight static and dynamic attributes based access control scheme for secure data access in mobile environment. In: Proceedings IEEE Local Computer Networks (IEEE LCN 2014), Edmonton, Canada (2014)

9. Pang, Z., Zheng, L., Tian, J., Kao-Walter, S., Dubrova, E., Chen, Q.: Design of a terminal solution for integration of in-home health care devices and services towards the Internet-of-Things. Enterp. Inf. Syst. **9**(1), 86–116 (2015)
10. Uckelman, D., Harrison, M., Michahelles, F. (eds.): Architecting the Internet of Things. Springer, Heidelberg (2011)
11. Weber, R.: Internet of Things – new security and privacy challenges. Comput. Law Secur. Rev. **26**, 23–30 (2010)