



HAL
open science

Toward a Smart Health-care Architecture Using WebRTC and WoT

Saad El Jaouhari, Ahmed Bouabdallah, Jean-Marie Bonnin, Tayeb Lemlouma

► **To cite this version:**

Saad El Jaouhari, Ahmed Bouabdallah, Jean-Marie Bonnin, Tayeb Lemlouma. Toward a Smart Health-care Architecture Using WebRTC and WoT. WorldCIST 2017, Apr 2017, Porto Santo Island, Madeira, Portugal. pp.531-540, 10.1007/978-3-319-56541-5_54. hal-01503434

HAL Id: hal-01503434

<https://hal.science/hal-01503434>

Submitted on 7 Apr 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Toward a Smart Health-care Architecture Using WebRTC and WoT

Saad EL JAOUHARI¹, Ahmed BOUABDALLAH¹, Jean-Marie BONNIN¹,
and Tayeb LEMLOUMA²

¹ Institut Mines-Telecom / TELECOM Bretagne
Network, Security and Multimedia department, Site of Rennes, FRANCE,
{saad.eljaouhari, ahmed.bouabdallah, jm.bonnin}@telecom-bretagne.eu
² University of Rennes 1, Lannion, France, tayeb.lemlouma@irisa.fr

Abstract. By coupling the real-time communication capabilities of WebRTC and the advantages brought by the Web of Things (WoT), we introduce the design of a new flexible healthcare architecture, in order to propose divers e-health services. In this ongoing work, we mainly focus on the services related to the remote healthcare of patients and elderly persons. We illustrate the interest of our architecture through the analysis of three main use cases: a remote and continuous monitoring of elderly peoples and a remote medical examination of patients and emergency intervention in case of an accident.

1 Introduction

In the last few decades, with the advances in the networking technologies and database systems, Ambient Assisted Living (AAL) researches have been applied in several domains in order to ensure end-users safety and quality of life (QoL) at home. As a result, several new concepts have been introduced such as personal health management and monitoring, health prevention, healthcare and welfare, in order to provide better medical services, in particular to the persons with special needs such as elderly and disabled persons.

The notion of e-health had emerged, where an enormous amount of health data and valuable medical knowledge have been electronically available and remotely accessible by different entities, specially through the Web. Hence it becomes easier to get, store, disseminate and exploit health information, especially with the improvement in the medical embedded sensors. Currently, the use of those medical Things is significantly increasing since they can continuously and rapidly provide medical information about the patient. Thus, we need an efficient way to manage all of those Things through a WoT/IoT architecture. Moreover, we need to define a way to transmit health data to the dedicated entity (Doctor, Hospital, etc.) in real-time.

In our vision of the e-health, we want to gain better relation between the doctor and the patient, where the doctor can access the medical Things of the patient in order to get a better understanding of the state of the patient. Hence

we propose a new concept where we benefit both from the advantages of the real-time communication, and the WoT/IoT. For instance, we propose the case of the remote continuous monitoring of disabled or elderly persons, especially in the context of an aging population.

To better understand the problems of this ongoing work, first we need to explain the main technologies that we will use all along this paper in section 2. Then, we will discuss the related works in section 3. Then, we present our new architecture in the section 4, and next we will show how we can use this architecture in a relevant use case of the smart health, in section 5. In section 6, we will demonstrate the implemented part of our ongoing work. Finally, we will discuss the future perspectives of our work.

2 Technologies

In this section, we will first introduce the technology used to establish the communication between the different users of this framework. Then we will introduce the notion of the Web of Things which, will enable us to control and interact with the different smart objects in our system.

2.1 WebRTC

As described in many RFCs, WebRTC overcame the challenge of converging into a single technology two traditionally opposite side of the Web represented by the asynchronous client-server paradigm on one hand, and the domain of the peer-to-peer multimedia and real-time communications on the other hand. Technically, it allows Web-based application (running in a browser) to exploit dedicate native APIs implemented in the browser. They allow the secure exchange of media and data in real time and in a peer-to-peer fashion [1, 2]. This technology is still under standardization by IETF and W3C. Currently, many browsers such as Google Chrome or Mozilla Firefox natively support WebRTC technology. It appears to be a very promising technology since it was estimated that it can have more than one billion endpoint in 2013³. In addition to the powerful communication capabilities provided by WebRTC, security was considered from the beginning in order to guarantee authentication, confidentiality and integrity in the system.

WebRTC uses mainly three APIs. *MediaStream* is responsible of the access to the Input/Output devices and the creation of the media stream. *RTCPeerConnection* is responsible of the creation of a secure media channel via SRTP (Secure Real-time Transport Protocol); it also encompasses signaling, NAT traversal and other important tasks. Finally *RTCDataChannel* allows the creation of a secure data channel via SCTP (Stream Control Transmission Protocol) encapsulated in DTLS (Datagram Transport Layer Security). One of the advantages of WebRTC compared to the other real-time communication technologies, is that the security have been taken into account from the beginning, by providing a strong security architecture [1, 2].

³ <http://io13webrtc.appspot.com/#10>

In addition to the basic functionalities of WebRTC, sending media and data in real-time, we think that we can extend it to a whole new level of innovation. This innovation can be achieved by opening for WebRTC applications a door to the Web of Things.

2.2 Web of Things

The Web of Things (WoT) is a specialization of the Internet of Things (IoT). In one hand, all the complexity of the connectivity part of the smart objects (or Things) is abstracted. In the other hand, the WoT provides a standard application layer based on Web standards to simplify the creation of IoT applications. In the IoT, the one-application one-communication protocol overwhelms, which creates silos of users and which does not present the full potential of the IoT. One of the main interest of using the WoT instead of the IoT is the different advantages it offers. Just to mention, the simplicity of development, the loose coupling, since HTTP is loosely coupled by design, and the openness of the standards. The idea is that all the Things can communicate using a Web language by exposing a REST API. This API can be either present in the Thing itself or in an intermediary that can act on behalf of the Thing to expose the Web API[3]. This has become possible with the improvement of the embedded systems fortunately. Nowadays, we can run tiny servers inside the constrained devices such as `lighttpd`[4] and `Nginx`[5].

3 Related works

The literature provides several examples of Web-based infrastructures aiming at improving the management of patients diseases. Some of them are based on old communication technologies such as in [6] [7] [8] [9]. We present below a summary of the main solutions that uses WebRTC.

We will focus on recent solutions some of which use WebRTC. We provide below a comparative analysis of some representative works.

A recent work [10] uses in addition to the audio/video communication part of WebRTC, the `DataChannels` to exchange medical data between different medical entities. Moreover the patients have access to medical sensors. In [11], the authors propose an e-health platform, providing specific services to the diabetic persons (Type 1 diabetes mellitus (T1DM)). The proposed solution uses medical sensors and a humanoid robot. The robot interacts/dialogs with the patient in order to get better information on his state. All of the data (sensors and dialog) are collected by the robot, and then sent to the health carers. The work of [12] presents an extension to WebRTC to enable peer-to-peer exchange of sensor data, and a proposal to enable Web applications to access sensor data and to bring nearby sensor streams to Web applications and multimedia communication over the Web. They mainly propose to extend the `MediaStream` API of WebRTC in order to manipulate also the sensor streams and to provide them to the customers. The approach is only theoretical without any implementation. In [13],

the authors present a video conferencing system allowing online meetings between remotely located care coordinators and patients at their homes. They use a special device (TeleMedCare) to monitor the vital sign of the patients. In this work, there is no interaction with medical sensors apart from the TeleMedCare.

4 General architecture

To develop the full potential of WebRTC, we propose innovative usages, which helps to extend its limits. It consists in adding new functionalities to control, manage and send data from/into the Things available in the WebRTC endpoint surroundings. We want indeed during a WebRTC multimedia session to be able to access the Things and to send data to the other side of the communication in real-time. This enables us to create new opportunities, since in this case, we consider that each WebRTC endpoint is a gateway to his own Things. Moreover, we can delegate the access to other persons for instance the remote person with whom we are talking. We think also that the security and the privacy aspects are very important, so we will dedicate particular interests in the problems that can occur inside our architecture.

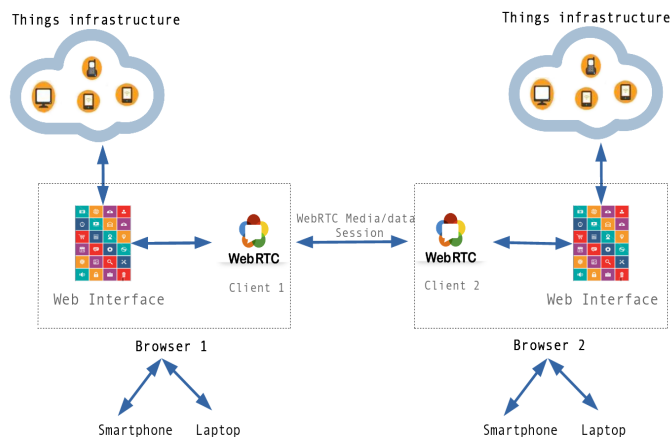


Fig. 1: Proposed architecture

A simplified view of the architecture of our platform is shown in Figure 1. Using a WebRTC compatible browser, a user can, in one hand, communicate in real-time, and in the other hand, access his Things. Accordingly to our WoT approach, the client application should both take care of the communication and the access to the Things. We suppose that each Thing is identified using a URI and offers a Web API which allows us to access the Things resources. The Things are located in the private network of each user (his home, office, ...).

We can use this architecture in e-health, as shown in figure2, where the users can be the patient/elderly/injured person in one endpoint of the WebRTC session, and the remote medical support, such as doctor/medical relief center/hospital, in the other endpoint. The Things represent the different medical embedded devices. Finally, we have the different stakeholders that can interact directly with the patient/elderly such as nurses, paramedics, ... They also have the authorization to access the personal medical devices of the person.

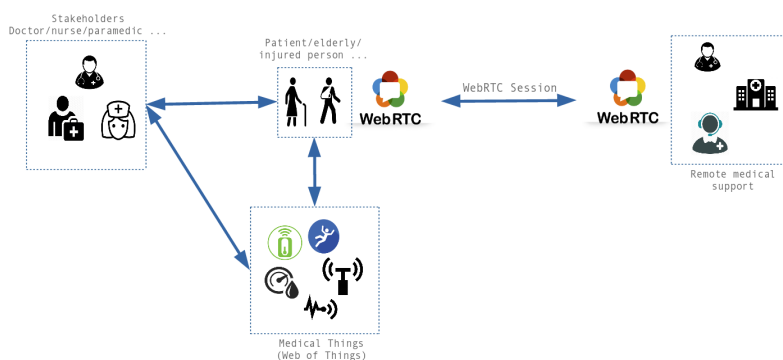


Fig. 2: Adapted architecture for e-health scenarios

5 e-Health Scenarios

To gain a better understanding of the idea behind this proposed architecture, we will illustrate it with a set of scenarios applicable in the field of the smart health (or e-health).

5.1 Continuous monitoring

The first use case is for continuous monitoring of a patient or an older person. In this case, wearable devices (such as a falling sensor, pressure sensor, peacemaker, ...) are continually monitoring the state of the patient. In case of abnormal behavior (falling, heart problem, ...), an alert is sent to the emergency relief center or to a doctor using WebRTC. In this case, we suppose that the WebRTC session is either always set on or established urgently, as shown in Figure 3. We can also imagine the **mobility** use case, where we continuously monitor the health status of a mobile patient. The patient is wearing embedded devices that continuously send health information to the medical center. This can be very useful to be able to detect abnormality in the comportment of the patient and to quickly provide medical help in the extreme case. We can also try to detect early syndromes of the most known diseases such as early detection of diabetes, AIDS, Alzheimer ...



Fig. 3: Continuous monitoring of a falling

5.2 Remote medical consultation

The case of a remote consultation with a doctor can be applied to disabled persons or old persons that cannot visit a doctor. We suppose in this case that, the patient has access to a set of medical devices. So, the patient starts a multimedia conversation with his remote personal doctor. At a certain stage, the doctor wants to know some health information about the patient. The patient puts on the wearable devices, accessible through the platform, and then sends the information received from those devices to the doctor. With this information, the doctor with his monitoring window showing all the information in real-time, can give advices or prescription, as shown in Figure 4. We can also have an emergency relief center (ERC), instead of a doctor. In this case, the self quantifying sensors of the patient, and the medical sensors prescribed by a doctor, do not have to transmit their data permanently to a medical center. This alleviates the deployment constraints regarding availability of the network and legal constraints regarding the storage of private data. The patient remains master of his data produced by the different sensors and decides when and with whom they can be shared.

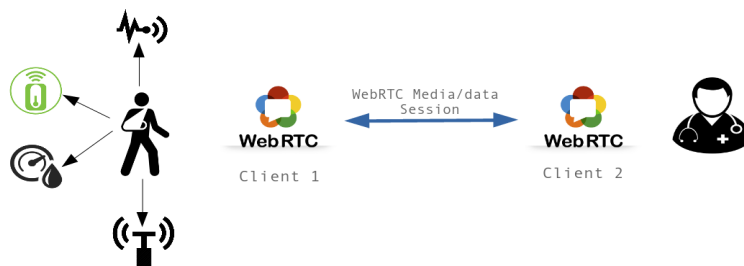


Fig. 4: General case

5.3 Emergency intervention in case of an accident

The last use case is the one of an accident, as shown in Figure 5. Once a paramedic arrives to the place of the accident, to help the injured person. First she/he will attach all the available wearable medical devices to the body of the person. We suppose that the paramedic had already an equipment that speaks WebRTC with a doctor or with an ERC that implements our architecture (capable of accessing smart objects in this case we consider the case of medical wearable smart objects). Then, the paramedic starts a communication with the remote ERC/ doctor, and at the same time the wearable objects will start sending vital information in real-time to the ERC, using WebRTC Datachannel. The paramedic will at the same time apply first aides to the person in the simple case, and in case of serious injuries the ERC/doctor can give real-time instructions, while transporting the patient. Moreover, the ERC can also contact the different hospitals to see availability for receiving the injured person and to start preparing a room for surgery, if the person needs emergency intervention.



Fig. 5: Emergency intervention in case of an accident

6 Implementation

Currently, we implemented a platform using Node.js. This platform can in one hand, establish a multimedia communication between two users just using their browsers. On the other hand, it enables users to access a set of predefined Things in his possession. Each user can then retrieve data from the different Things and send them to the remote user in real-time using WebRTC Datachannel. The figure 6 shows that a user can access an accelerometer and an ultrasound sensor. Such an approach can be nearly extended to all kinds of Things.

The first part of the architecture is WebRTC, it is composed of two clients, a signaling server and an Identity Provider (IdP). First the client gets the WebRTC JavaScript application from the server, and execute it inside the sandbox of the browser. Then, the signaling process where the user reaches/contacts the remote part of the communication. Finally, with all the exchanged parameters, a WebRTC media/data channel can be set. As for the IdP part, it provides the

users with an identity (after a valid authentication on the IdP side), so that they can authenticate themselves to the other side of the communication.

The second part is the WoT part. Each endpoint is considered as a gateway to his set of Things. Also, each user has a control on those Things and only his can access the respective Things. In our case, the user communicate with the less constrained Things using HTTP, and using HTTP with an intermediary in case of constrained Things. The intermediary can then communicate with the constrained Things using one of the well known communication protocols (Zigbee, Bluetooth, WIFI ...). In our current platform, we have an accelerometer (ADXL335) attached to a Raspberry Pi 2, and an ultrasound (HC-SR04) attached to an Arduino UNO, which send data to another Raspberry using WIFI adapted to the Arduino. The Raspberry used in our architecture is a Model B with ARMv6 single core, 700 MHz and 512 MB SDRAM. In this case, the Web API is implemented in the Raspberry and each Thing is identified by using a URI. Each user has access to an accelerometer and to an ultrasound sensor.

In the case of the e-health scenarios, all we need to have are wearable medical Things. They can communicate with any protocol, since the WoT abstracts all the complexity of the connectivity of the Things.

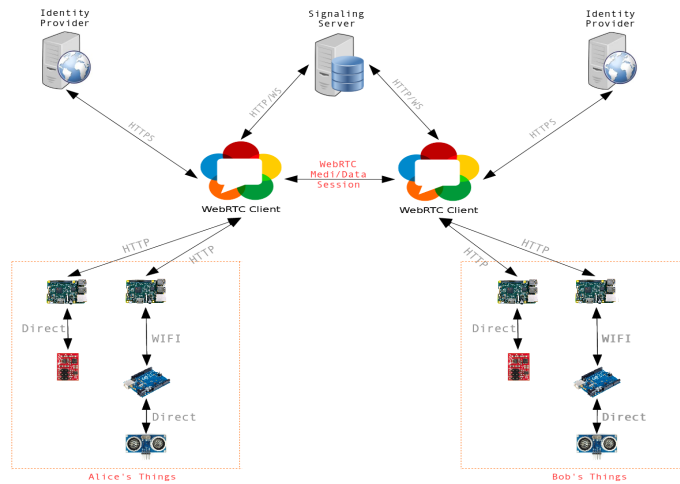


Fig. 6: Current Implementation of the platform

Using the current state of the architecture, an implementation of the remote consultation use case is possible, where a doctor is communicating with a patient using WebRTC. The patient has access to a set of Things. Then, the patient can send these information to the doctor in real-time using WebRTC's Datachannel. Finally, the doctor can monitor the received data and provide a prescription to the patient (this prescription can also be sent to the patient using WebRTC).

7 Conclusion and perspectives

The following paper present an ongoing work for the creation of an e-health framework specially conceived for remote examination of patients and injured persons in case of accidents. Though the different use cases, we demonstrated the advantages that can be brought by such framework, such as better and more efficient interaction with the medical entities, the continuous monitoring of the patients, the remote healthcare services and so on and so fourth. Finally we presented our current implementation of this framework. We also analyzed other use cases mainly related to the emergency intervention in case of an accident, and in the case of the mobility of the patient outside his habitual system (home, hospital, ...). However, due to the lack of space, we couldn't include them in this paper.

The next step of our work is to investigate of all the security and privacy aspects of this platform. Currently, the communication between the WebRTC clients and the different Things is not secured yet. We need first to evaluate the main WoT security mechanisms existing today. CoAP/DTLS [14] seems an interesting candidate to secure the communication with the different Things. The second aspect is the access control. Only the authorized persons should access the Things. We identified several possible solutions to control the access such as RBAC [15] and ACL [16]. The delegation of the access which opens interesting use cases, can be tackled with OAuth 2.0 [17]. However, in term of performance, OAuth is not very lightweight, so we consider this solution in the case of the use of a gateway. We are also interested to analyze the introduction of the notion of the profile of the patient [18] (the persons profile can include medical record, chronicle diseases, ...). Depending on this profile, the framework will use a defined set of medical devices to monitor the physiological status of the patient. This can be interesting in case where the patient is in a hospital, the nurse can install the wearable medical devices on the body of the patient, and according his profile, the corresponding Things will transmit the medical data to the framework. This includes also, managing the transmission frequency of the different medical devices in a given environment (smart-house, smart-hospital, smart-city ...), depending on its importance/priority[19], in order to optimize our system and to have better and secure management of the data. This priority can be accorded depending on the profile of the patient. We think that this approach can open the door to the development of new healthcare ecosystems that combine the communication technologies, the WoT/IoT and the medical expertises of the different medical entities, in order to provide quicker, better and more efficient medical intervention to the persons in needs.

Acknowledgment

For the first two authors, this work has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 645342, project reTHINK.

References

1. A. Sergiienko, *WebRTC Blueprints*. Packt Publishing Ltd, 2014.
2. C. Jennings, T. Hardie, and M. Westerlund, “Real-time communications for the web,” *IEEE Communications Magazine*, vol. 51, no. 4, pp. 20–26, 2013.
3. D. Guinard and V. Trifa, *Building the Web of Things*. Manning Publications Co, 2016.
4. lighttpd, “lighttpd,” 1993. [Online]. Available: <http://www.lighttpd.net/>
5. nginx, “nginx.” [Online]. Available: nginx.org
6. F. Magrabi and al., “Home telecare: system architecture to support chronic disease management,” in *Conference Proceedings of the 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 4, 2001.
7. C. Lau and al., “Asynchronous web-based patient-centered home telemedicine system,” vol. 49, 12 2002.
8. H. Zheng, R. J. Davies, and N. D. Black, “Web-based monitoring system for home-based rehabilitation with stroke patients,” in *18th IEEE Symposium on Computer-Based Medical Systems (CBMS’05)*, June 2005, pp. 419–424.
9. C. Y. Chiang and al., “An efficient component-based framework for intelligent home-care system design with video and physiological monitoring machineries,” in *Fifth International Conference on Genetic and Evolutionary Computing*, Aug 2011, pp. 33–36.
10. P. Pierleoni and al., “An innovative webrtc solution for e-health services,” in *IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom)*, Sept 2016, pp. 1–6.
11. M. A. Al-Taeae and al., “Web-of-things inspired e-health platform for integrated diabetes care management,” in *IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT)*, Dec 2013, pp. 1–6.
12. J. a. Azevedo, R. L. Pereira, and P. Chainho, “An api proposal for integrating sensor data into web apps and webrtc,” in *Proceedings of the 1st Workshop on All-Web Real-Time Systems*, ser. AWeS ’15. New York, NY, USA: ACM, 2015, pp. 8:1–8:5. [Online]. Available: <http://doi.acm.org/10.1145/2749215.2749221>
13. J. Jang-Jaccard and al., “Webrtc-based video conferencing service for telehealth,” *Computing*, vol. 98, no. 1, pp. 169–193, 2016.
14. Z. Shelby, K. Hartke, and C. Bormann, “The Constrained Application Protocol (CoAP),” RFC 7252, Internet Engineering Task Force, Jun. 2014.
15. R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, “Role-based access control models,” *Computer*, vol. 29, no. 2, pp. 38–47, Feb 1996.
16. C. M. Medaglia and A. Serbanati, *An Overview of Privacy and Security Issues in the Internet of Things*. New York, NY: Springer New York, 2010, pp. 389–395.
17. D. Hardt, “The OAuth 2.0 Authorization Framework,” RFC 6749 (Proposed Standard), Internet Engineering Task Force, Oct. 2012.
18. H. Mshali, T. Lemlouma, and D. Magoni, “Analysis of Dependency Evaluation Models for e-Health Services,” in *Proceedings of the IEEE Global Communications Conference*, United States, Dec. 2014, p. 7 pp.
19. —, “Context-aware adaptive framework for e-health monitoring,” in *IEEE International Conference on Data Science and Data Intensive Systems, Sydney, Australia*, Dec 2015, pp. 276–283.