Confidere – A Trust Model for Context-Aware System on HealthCare Domain

[1]Marcelo Henrique Monier Alves Júnior, [2]Francisco José da Silva e Silva, [3]Luciano Reis Coutinho, [4]Samyr Vale, [5]Lucas Maia
1, First Author *Federal University of Maranhão, Brazil, marcelo.alvesjunior@ifma.edu.br*
**2,Corresponding Author* *Federal University of Maranhão, Brazil, fssilva@deinf.ufma.br*
3,4,5 Federal University of Maranhão, Brazil

Abstract

Mobile health is the name given to the practice of medicine and health care through mobile device applications. This work is under the scope of the MobileHealthNet project, which provides a context-aware middleware focused on creating health care applications in Mobile Social Networks (MSNs). MSNs are collaborative environments with large variety of different contexts. One of the drawbacks in the MobileHealthNet is the absence of Quality of Context (QoC) between consumers and producers of data, which can cause problems, such as wrong decisions by physicians, due to data collected from a sensor with low accuracy. This paper focuses on implementing QoC parameters in the MobileHealthNet infrastructure. Our main goal is to provide support to Trustworthiness, then we suggest a trust model named Confidere to detect the trust of Context Provider (CP) component. Finally, we evaluate Confidere in occurrence of CP's malicious behavior in different scenarios.

Keywords: *Middleware; Mobilehealthnet; Trustworthiness; Quality of Context; Confidere.*

1. Introduction

The increased use of mobile devices has demanded practical applications for these devices in several areas of human knowledge and practice, such as health care, education, entertainment, financial system, e-commerce, and others [3]. Essentially, the aim is to promote the access to information as easily as possible from anywhere and at any time provided by the use of these devices and applications. One important factor contributing to this phenomenon has been the decrease in the cost of devices and the increase in their computing resources. These reasons boost the development of robust applications, particularly ubiquitous applications, in academic and industrial areas.

The term "ubiquitous computing" was defined by Weiser [1] as a human-computer interaction model in which one seeks to improve the use of computers through the provision of numerous devices interacting with one another and with the users in a transparent manner. Ubiquitous applications are transparently adaptable according to the user's environment, and they do not require explicit user intervention.

mHealth applications enable health care professionals for monitoring patients not requiring a same physical space. In other words, it is possible that physicians, nurses and other health care professionals for remotely treating and monitoring their patients through the use of these devices.

Considering this demand for mHealth, the Mobile Social Networks for Health Care in Regions Offside (MobileHealthNet) project was created, consisting in building a middleware to assist the development of applications in the health care domain. Through this middleware is possible, for example, establish a social network mobile from which health professionals, patients and the wider community can exchange information and experience about a particular treatment, medications, health campaigns, etc. Therefore, the health monitoring can be remotely performed, and multidisciplinary teams in which professionals from different specialties may help in treating patients.

The middleware also provides a large collaboration among professionals of various health care areas, in addition to a better information flow among them; improvement in information of patient, which

contributes to the therapeutic process and the increase in decision making quality on the treatment of patients[3].

Another feature of the project is that it provides support for Quality of Context (QoC). QoC is any information that describes the quality of information that is used as context information [4].The QoC consists of parameters and the project under discussion, provides four of them, namely: Freshness, Accuracy, Frequency and Refresh Rate.

The main objective of this paper is the implementation of a specific QoC parameter called Trustworthiness in the context service layer of the MobileHealthNet.

This layer is responsible for storing and providing the context information. Different context types may be operated in m-health applications, e. g, the patient location information that requires emergency assistance.

In this context, it is essential to implement QoC in applications developed with the help of middleware's resources, because the context information trafficked between users, requires a high level of quality. The high level of quality is very important, because it is directly related to decision-making by health professionals, which in turn direct impact on the health status of their patients. For example, the Trustworthiness in components is highly significant for m-health applications because their behavior. In an example, the decision making of a physician, can become wrong due to incorrect information provided by an uncalibrated sensor of mobile device.

According to Russell [5], trustworthiness is related to the entity receiving the trust, in other word, the Trustee, as the entity that will provide the trust is entitled as Trustor. Trust is the expectation that the Trustor will have about the Trustee behavior, i.e., if the Trustee will behave as he considers trusted.

This paper is organized as follows. Section 2 discusses trust computation. Section 3 describes the MobileHelathNet project. Section 4 investigates related work. Section 5 describes a trust model and the implementation of the proposed system. Section 6 evaluates the system performance. Finally, Section 7 draws the conclusion.

2. Trust Computation

According to Bezemer [6], the psychology describes confidence as a ratio of three parts: A trust B with respect to X, in which A represents who will have the trust (Trustor); B, who will receive trust (Trustee); and X refers to the matter on which is being established the trust. For example: the user (A) trust on sensor (B) to provide him the location (X) at the time it is stopped. However, do not trust on it to provide you with the same information when it is in motion.

Another way to establish trust is by reputation. This is provided by third-party opinion, i.e., the confidence given to trustee will depend upon information passed to Trustor by third parties who have previously interacted with it.

2.1. Trust Computation

According to Russell [5], the trust condition may be seen as the Trustor waiting for a Trustee to be faithful. The degree of trust is a set of Trust conditions to see if the Trustee will have more or less trust in a particular subject.

2.2. Building trust

The trust is given to the other through prior knowledge or through a record interactions. Thus, it is impossible to establish confidence without interactions between the parties, because as we know, trust needs to be built. However, the literature discusses three initial interactions between individuals:

generalized trust, control, and reputation. The first is used the risk of harm if it is too low, in which case the Trustor will have to believe in trustworthy behavior of the Trustee; the control is used when the Trustor can prevent the action of the Trustee to betray trust, e.g., by monitoring and providing arrangements of their behavior and ultimately the reputation that is provided through third-party information [7].

2.3. Trust management

In accordance with Jøsang [8], trust management is the activity of creating systems and methods that enable trust parties to evaluate and take decisions regarding the reliability of transactions involving some potential risks, also allowing users and system owners can increase and correctly represent the reliability of themselves and their systems.

3. MobileHealthNet

The MobileHealthNet project architecture has a layered structure, as can be seen in Figure 1. The first layer is the Communication layer, called MobileHealthNet Communication Framework, which contains all the mechanisms to facilitate data sharing in Mobile Social Networking (MSNs). It is based on the Scalable Data Distribution Layer (SDDL) [9], that adopts two communication protocols: (i) the Mobile Reliable UDP (MR-UDP) [10], used in all communications between mobile devices and the core services running in a cloud computing infrastructure; and the (ii) Data Distribution Service (DDS) from the Object Manage Group (OMG), used for communication between servers that run the middleware core services in the cloud.

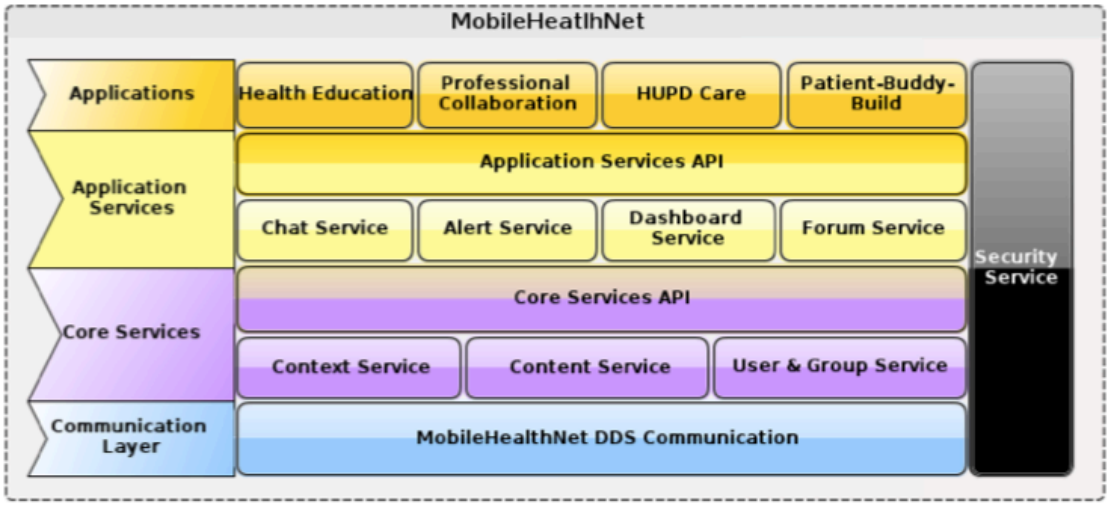

Figure 1. **MobileHealthNet Architecture**

The next layer is the Core Services, through which basic services are made available to applications developed with the MobileHealthNet middleware. This layer is composed of three services:

- **Context Service:** offers services for gathering and distribution of data context. The trust evaluation model proposed in this work is implemented in this layer;
- **Content Service:** responsible for storing media (images, video, audio, text, etc.) posted by the MSN users. This service allows the tagging of each media with application defined information (for example, the kind of disease that the content is related with, such as asthma);
- **User & Group Service:** manages the MSN users and groups.

The application service layer provides a set of services that are typical to social networks, such as an alert service (for distributing users notifications), chat, forum, and a publishing service for posting content on murals. The Security Service layer is transversal to all layers of the MobileHealthNet

architecture, and provides mechanisms that implement the defined security and privacy model. Finally, there is the Applications Layer, which provides already developed applications that use the services provided by the middleware.

3.1. MobileHealthNet Context Service

Figure 2 summarizes the structural aspects of the MobileHealthNet context service architecture, named MobileHealthNet Context Service (MHNCS). This architecture is an extension of the Context Management Service (CMS), described in [10].

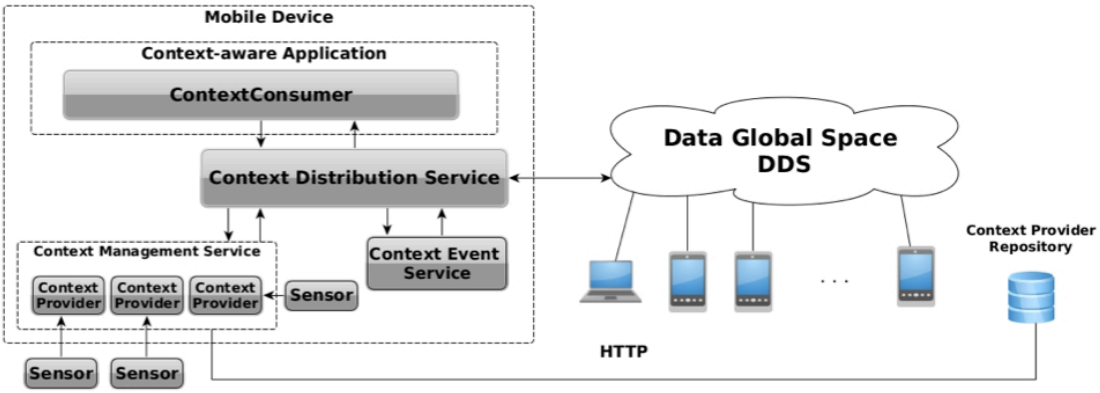

Figure 2. **MHNCS Architecture [11]**

As noted in Figure 2, the MHNCS consists of the following basic components:

1. The Context Providers (CP) are responsible for gathering data from sensors. Each CP is responsible for collecting data from a specific sensor;
2. Context aware applications specify which type of context data they require for executing and the MHNCS Context Management Service (CMS) checks their availability by scanning the user nearby sensors. It also dynamically downloads and instantiates CPs from an external repository [11].
3. The Context distribution Service (CDS) is responsible for managing the distribution of context data for customers who have registered interest in receiving them; and
4. A Context Event Service (CES) provides a mechanism to evaluate the occurrence of events related to the user context data. Applications register events of their interest, that are defined by an expression. For instance, the expression heart rate > 160 will trigger an event notification when a sensor that the user is wearing indicates that his/her heart beat rate exceeds 160 beats per second.

4. Confidere

In Section 2 we exposed the need for a high level of trust in the middleware components and the information generated by them. The proposed work has the objective to evaluate the trust level of a particular component of the MobileHealthNet infrastructure: the CP. This component is responsible for gathering all context information required by the user applications. The Confidere application will evaluate the confidence level of each CP used by the user applications. Thus, the Trustor will be the Confidere application and the Trustee the CP.

This paper proposes a trust model for context aware systems. A system developed from the model evaluates the CP behavior at runtime based on its use of the mobile device resources. One of the main problems is how to ensure trust in CPs that are downloaded from repositories located on remote servers and that are instantiated dynamically. To better explain our approach, we divided it into two parts: the mathematical representation of trust assessment (how the CP level of trust is calculated) and the application developed as a proof of concept.

4.1. Mathematical Representation

The trust model defined in this paper is based on a work of Russell [5], in which he states that trust is given by the Trustor based on the behavior exhibited by the Trustee. However, this behavior will not be considered reliable if the Trustee does not behave in a way expected by the Trustor. Another determining factor in the model is the concept of reputation, in which trust is achieved through third-party opinions.

The first type of trust is obtained by direct interactions, or transactions that the Trustor had with the Trustee. At each new interaction with the Trustee, the Trustor will have a different level of trust. The reputation is achieved through indirect interactions, i.e. the direct interactions that others have had with the Trustee.

In (1), *D* is the direct interactions; *a* and *b* represent the client application (the Trustor) and the context provider (Trustee) respectively; *c* is the context in which *b* is evaluated, this context consists of properties, which in turn are represented by the maximum amount of CPU usage, RAM memory and Bandwidth that *a* thinks that *b* must use to be considered trustowrthy; *i* represents the interaction number. If *a* considers that *b* is trustworthy in the context *c*, a point to the current value of trust *b* is added, otherwise it is taken one point from the current value of his confidence.

$$D_{ab}^{c}(i) = \begin{cases} +1, \ if\ a\ trust\ in\ b \\ -1, \ otherwise \end{cases} \quad (1)$$

The total amount of direct interactions is given by the (2), in which, T_{ab}^{c} represents the sum of direct interactions that there were between *a* and *b* in context *c*.

$$T_{ab}^{c} = \sum_{i=1}^{n} D_{ab}^{c}(i) \quad (2)$$

The value of the indirect interaction, or the reputation of *b*, is obtained by (3),

$$R_{b}^{c} = \frac{\sum_{i=1}^{k} T_{ib}^{c}}{k} \quad (3)$$

in which, R_{b}^{c} represents the average of all interactions made by *b* in the context *c*; *k* is the amount of users who interacted with *b* in the context *c*.

Total trust is given in (4). It refers to the sum of the direct and indirect interactions,

$$C_{ab}^{c} = T_{ab}^{c} + R_{b}^{c} \quad (4)$$

wherein, C_{ab}^{c} corresponds to the final trust value *a* to *b* will in context *c*. As we can see in (4), trust is given to an entity through prior knowledge or through recorded interactions. Thus, it is impossible to establish trust when there has never been any interaction, because, as we know, trust needs to be built.

4.2. Confidere Application

Confidere is an application developed to provide trust management to the Context layer of the MobileHealthNet middleware. It serves to assess the level of trust of CPs, that is, to detect whether a CP will behave in the manner expected by the client application.

The application architecture is shown in Figure 3. It consists of the following components: i) the Select Best Context Provider (SBCP), responsible for selecting the Context Provider that best fits the requirements of the client application; ii) the Monitor Context Provider (MCP), responsible for monitoring the CP execution; iii) the Evaluation Context Provider (ECP), responsible for evaluating the CP, that is, the component that calculates the trust level of the CP.

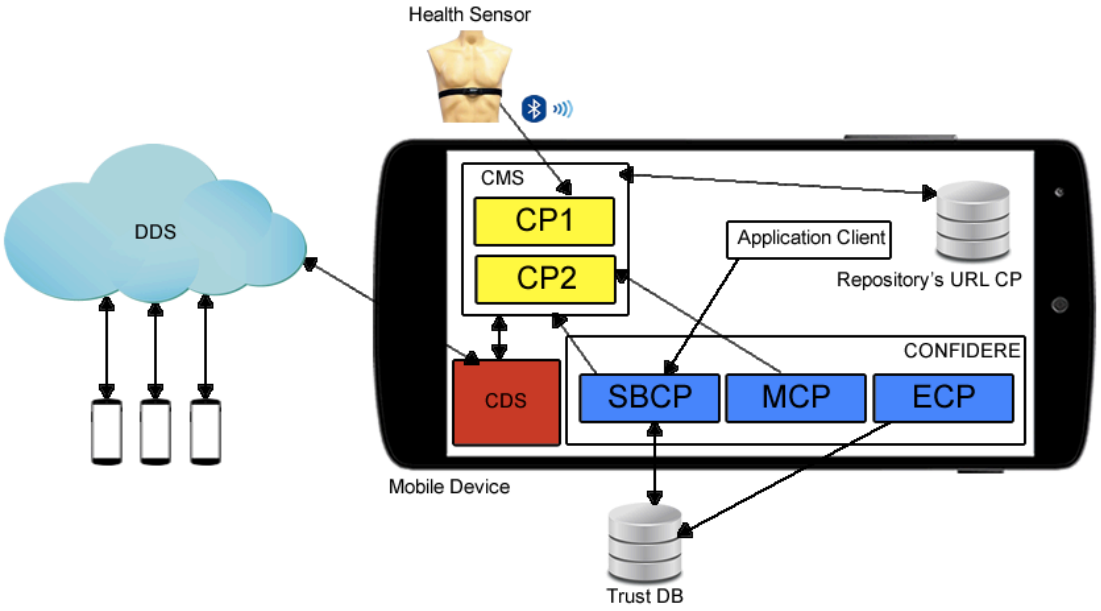

Figure 3. **Confidere Architecture**

Besides Confidere components, there are other elements which help to maintain a high level of trust in CPs. For example, the Trust_DB is a remote database responsible for storing the trust levels and characteristics of CPs. Another component of this architecture is the CMS, the element responsible for downloading and managing all Context Providers. Finally, a local database is responsible for storing the addresses of the CPs.

To better illustrate the use of the Confidere application, we developed a sequence diagram. Figure 4 includes all enforcement actions from the request of the CP by the client application up to the its disposal.

The client application requests to the SBCP component the required type of CP with the trust level that it should have and in what context it should be inserted. The context is represented by the set of properties of the CP, in other words, what are its resources permissions (contained in the file called *AndroidManifest.xml*) and how much of CPU, RAM and the Bandwidth the CP is supposed to consume. For instance, suppose that for *a* to consider *b* trustworthy, B should use only 5% of the CPU, 10% of RAM and 20 MB of bandwidth. Then, the component makes a query to Trust_DB that returns the Context Provider name that best fits the requirements of the requesting application. After that, the SBCP sends to the CMS the name of the CP, which in turn looks up the address of the CP in the local repository. Then, the CMS downloads and installs it.

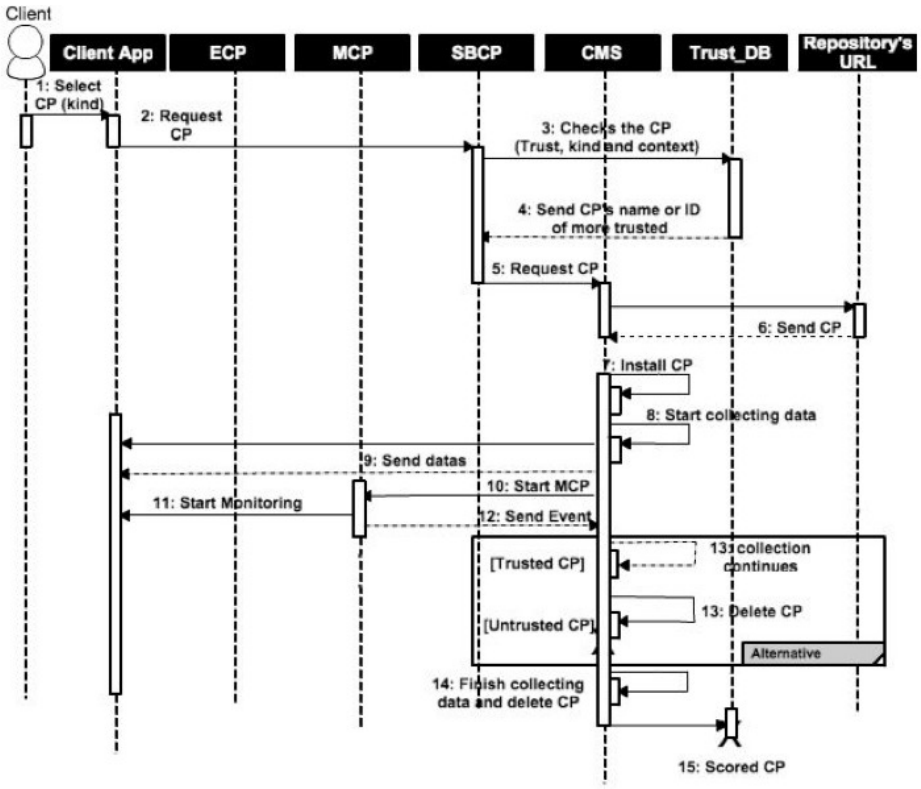

Figure 4. **Sequence Diagram**

Subsequently, the MCP component begins to monitor the CP. If the CP exceeds the values defined by the client application, the MCP removes a point of the CP trust level, otherwise it increments a point. When the CP is no longer required by the client application, the CP is deleted from the mobile device. However, if it is not trusted, it is deleted before the end of the sensor data gathering.

5. Evaluation

The proposed model aims to ensure the security of mobile terminals with respect to dynamically downloaded code. Confidere has three features: it selects the CP that is more trusted and suited to the demands of the client application, it monitors the provider's behavior and, therefore, scores the provider's trust level.

The main purpose of the evaluation is to determine the Confidere's ability to detect malicious CP, that is, if the CP behaves as expected by the client application and the time that it performs this function. For this purpose, we have developed CPs and applications that require these providers to receive user information from the environment.

For this evaluation, 9 CPs were developed, where six of these are not considered trustworthy in any context registered in the existing trust database (the Trust_DB). The information contained in the database were included through feedback from users regarding the use of the providers in their respective contexts. The other CPs are consistent with the requirements of the applications and are relied on at least one of contexts.

The applications developed for performing the Confidere evaluation follow in the domain of mobile social networking targeting health education. They allow patients and health professionals to share multimedia files (text, images, sounds, and videos) that are related to health care, exploring the concepts of social networking, where users can interact with each other commenting

or opining about any published content. The applications also allow to share the patient and health professionals' current locations, and provide features for finding the nearest hospitals or health care centers.

The applications developed for the evaluation have particular requirements for CPs. For example, the first application requires a Location CPs, uses only 10 to 15 percent of the CPU, 5 to 10 percent of RAM and from 5 to 30 mega of data traffic. Another requirement of the application is that the same provider only has permission to access the Internet and GPS. All of this information constitute a specific context, the context 1. The trust levels of these providers in each context are stored in Trust_DB.

Confidere evaluation experiments involved the monitoring of the following parameters: usage of CPU, RAM memory and bandwidth for each CP. If the CP exceeds the use of the resource specified by the requesting application, it has decremented a point from its trust level, meaning that it is entitled as unreliable. Otherwise, it is considered trustworthy.

Figure 5 shows the results of the evaluation of Confidere. Each CP had a time of 1 minute and 20 seconds to run. If during this execution interval there was no inappropriate behavior by the CP, it is finalized and considered trusted. Otherwise, the CP is finalized at the time of detection of an inappropriate use of the resource, and its trust is decremented.

The contexts were divided as follows: i) 1, 2, 3: specific to location providers; ii) 4, 5 and 6: specific to battery providers; iii) 7, 8, 9: specific to time providers. We can to see that during the evaluation process, in the time of 1' 20", the Confidere obtained 100% of success in detecting the trust level of the CPs, in accordance with the requirements of the applications. However, we can not say that the Confidere will always detect accurately or long time, the CP's trust level.

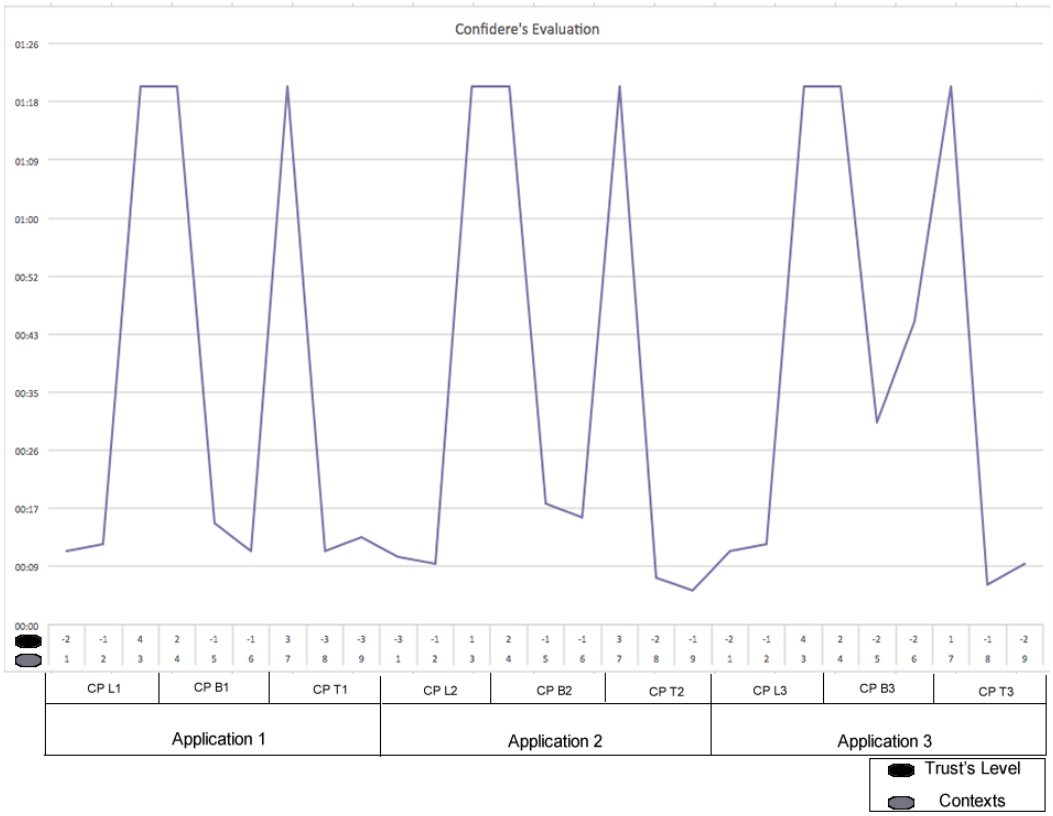

Figure 5. **Confidere's Evaluation**

For example, we have the location CP of the application 1, where in Confidere can detect the malicious behavior of the CP in the contexts 1 and 2 in 09 and 11 seconds, respectively. In relation to the context 3, for that the Confidere asserts that it is trusted, it is necessary that the estimated time for running its, comes to an end.

Another fact that should be highlighted, is that the Confidere takes more time to detect the untrusted Battery CPs. This is due to the way the CPs were written, since the CPU and RAM overload only occurs a few seconds after the data gathering is started. This is different from the others CPs (Location and Time), since in them the CPU and RAM overload occurs in parallel with the process of data gathering.

6. Conclusion

In this paper, a new model was proposed to manage the trust in context-aware system, with main focus on MobileHealthNet project. It supports different aspects of trust and it is able to calculate the trust level according to the parameters that the Trustor requires. It is flexible in relation to other existing models in the literature, because the properties are reported by the client application. It decides what features of mobile devices and how much of each CP can be used.

This model represents a substantial gain in quality of the components responsible for collecting and transmitting data of patients with chronic diseases, it is essential to the relationship of trust between physicians and system, as the level of trust directly influences the decision-making taken by the doctor or health professionals involved.

After reviewing, we realized that the Confidere could detect, in exposed time, the improper behavior of CPs. Thus, the application developed based on the proposed model can ensure if the CP analyzed is trusted or not.

Acknowledgment

The authors would like to thank FAPEMA (State of Maranhão Research Agency) and the Computer Science Postgraduate Program at UFMA (Federal University of Maranhão) for the support of this work.

10. References

[1] M. Weiser, “The computer for the twenty-first century,” Scientific American, vol. 9, Sep. 1991, pp. 94–100, accessed: 2015-03-30. [Online]. Available: http://www.ubiq.com/hypertext/weiser/SciAmDraft3.html
[2] M-Health: Emerging Mobile Health Systems. Springer Science Business Media, Inc., 233 Spring Street, New York, NY 10013, USA, 2006.
[3] A. S. Teles et al., “Mobilehealthnet: A middleware for mobile social networks in m-health.” Paris, FR: Springer, November 2012.
[4] T. Buchholz and M. Schiffers, “Quality of context: What it is and why we need it,” in In Proceedings of the 10th Workshop of the Open View University Association: OVUA’03, 2003, pp. 35–42.
[5] R. Hardin, Trust and Trustworthiness, R. S. Foundation, Ed. Series on Trust, 2002.
[6] D. Bezemer, “Trust: Forms, foundations, functions, failures and figures, ”Journal of Behavioral and Experimental Economics, vol. 34, no. 3,2005, pp. 421–424.
[7] J. E. Stark, “Trust in distributed computing,” Master’s thesis, The University of Guelph, 2014.
[8] A. Jøsang, C. Keser, and T. Dimitrakos, “Can we manage trust?” in Proceedings of the Third International Conference on Trust Management, ser. iTrust’05. Berlin, Heidelberg: Springer-Verlag, 2005, pp.93–107.

[9] I. Vasconcelos, R. Vasconcelos, G. Baptista, C. Seguin, and M. Endler, "Developing applications tracking and mobile communication using middleware sddl," in Salão de Ferramentas, Brazilian Symposium on Computer Networks and Distributed Systems (SBRC 2013), 2013, pp.1084–1091.
[10] L. Silva, M. Endler, and M. Roriz, "Mr-udp: Yet another reliable user datagram protocol, now for mobile nodes," Monografia em Ciência da Computação, vol. 6, 2013, pp. 40–43
[11] D. N. Pinheiro, "Mhncs: A middleware for the development of context-aware mobile applications with qoc requirements," Master's thesis, Universidade Federal do Maranhão, 2014.
[12] M. Malcher, J. Aquino, H. Fonseca, L. David, A. Valeriano, and M. Endler, "A middleware supporting adaptive and location-aware mobile collaboration." in Mobile Context Workshop: Capabilities, Challenges and Applications, Adjunct Proceedings of UbiComp, 2010, pp.418–424.
[13] R. Neisse, M. Wegdam, and M. van Sinderen, "Trustworthiness and quality of context information," in 9th International Conference for Young Computer Scientists, ICYCS 2008. Los Alamitos, CA,USA: IEEE Computer Society Press, November 2008, pp. 1925–1931,accessed: 2015-02-25. [Online]. Available: http://doc.utwente.nl/65203/
[14] R. Neisse, M. Wegdam, and M. van Sinderen, "Context-aware trust domains," in Proceedings of the First European Conference on Smart Sensing and Context, ser. EuroSSC'06. Berlin, Heidelberg: Springer-Verlag, 2006, pp. 234–237.
[15] F. Almenárez, A. Marín, C. Campo, and C. G. R., "Ptm: A pervasive trust management model for dynamic open environments," in FirstWorkshop On Pervasive Security, Privacy And Trust Pspt'04 In Conjuntion With Mobiquitous, 2004.