Context-Aware Security Using Internet of Things Devices

Michal Trnka[1(✉)], Martin Tomasek[1], and Tomas Cerny[2]

[1] Computer Science, FEE, Czech Technical University, Technicka 2, Prague, Czech Republic
{trnkami1,tomasma5}@fel.cvut.cz
[2] Computer Science, Baylor University, One Bear Place #97356, Waco, TX 76798-7356, USA
Tomas_Cerny@baylor.edu

Abstract. Current trends aim to extend software applications with context-awareness. Nowadays, there are already various approaches enabling security based on context, unfortunately there have limitations. However, the challenging topic is how to obtain as much context information about user as possible. Current progress in Internet of Things domain could be leveraged to obtain more context data. We propose a method to formalize context based on Internet of Things devices and use it for application context-aware security. Our approach is based on composition of a tree topology correlating to the user's devices for recurring situations. Based on changes in the tree we determine unusual behavior, trigger events or invoke specific actions.

1 Introduction

The emerging amount of mobile technologies [4], as well as the growing users' demands for personalized applications provide a base for current trends moving software applications towards context-awareness (CA) [1, 6]. Applications provide personalized content based on user's context or the application's context [5]. This brings novel experience to the applications users. However, securing applications is usually done the traditional way, assigning users various application roles, permissions for resources or security rules independent to the context. There are only few applications having the security based on context information. Nevertheless, we can expect that users and application owners would take the advantage of application security that uses context to provide specific resource control.

Applications using Context-Aware Security [11] (CAS) can be much less obtrusive for users. They can be asked for different authentication methods based on context. The result of the authorization for specific resource may vary depending on their context. For example, access from City A can have different access rights then access from City B. They can even sometimes omit authentication because their context is trustworthy by itself (e.g. access from inner company network). The context even could be created based on devices that a user uses. Each device has unique ID and it communicates with another devices that also communicate with another devices, therefore their interaction and position could be used to create user context. Similar to users, also application operators can profit from the context-based authentication. Different application might define stricter security rules for suspicious users' behavior (e.g. Internet access to

K. Kim and N. Joukov (eds.), *Information Science and Applications 2017*,
Lecture Notes in Electrical Engineering 424, DOI 10.1007/978-981-10-4154-9_81

system's confidential resources at night). The usage of context allows system administrators to manage more fine-grained security rules, which would otherwise tangle through multiple rules and make them unsustainable for maintenance. Another advantage is that system may automatically flag suspicious users and prevent them from doing certain actions.

However, the problem is how to obtain context from the user. Some information is obvious for the system (e.g. time, frequency of log-ins, history of application-user communication), other can be guessed but not guaranteed (e.g. geographical location determined from the IP address) while a lot of information are difficult to obtain (e.g. biometric information about the user). All of those mentioned information about the user's context may significantly increase security of the system, while significantly improving the application's user experience. In the following pages we will describe our approach to the issue by involving Internet of Things(IoT) devices to obtain user's context.

This paper is organized as follows: The following section describes related work, followed by our promising solution. The solution is demonstrated in case study section and the paper ends with conclusion remarks.

2 Related Work

Kranz et al. [8] describes the general interaction of the IoT devices with people. It focuses on few use cases with various augmented objects to verify that those areas are suitable for the concept of IoT interaction and that there are benefits. The results of this work indicate that certain areas of the IoT interactions are repeating in all scenarios, while some are unique. Nevertheless, there is no conclusion (or even framework/method proposal) and the paper just states that IoT is promising solution for many areas of human activities.

Petriu et al. [9] discusses possibilities and usage of the sensor-based real-time applications using information from users. They propose multiple communication processes and management system for heterogeneous functions of such system. While there are numerous significant methods and proposals, there is none that would use user's context for application security.

Ho et al. [7] describes framework involving user's context in mobile devices to reduce the amount of communication from different devices. This work focuses more on timing of the messages and their aggregation. It uses innovative ways how to obtains user's context. However, security is not addresses in the paper.

Interesting way how to retrieve user's context is to integrate sensors with items of daily use. Farringdon et al. [12] describes the usage of wearables, especially jacket, to retrieve real-time information for context awareness. The methods described in the paper addresses very well context retrieval, but it does not discuss the further usage of the context.

Context-aware security architecture for next generation applications is well described by Covington et al. [10] in his research. It describes all advantages of the context usage as well as its implementation. It only uses basic context that can be

obtained about the particular user through the application. Therefore, the context information is very limited and does not provide the big picture about user.

Another method for context-aware security describes Hu et al. [11]. This work proposes extending the role-base access control [13] model with context aware elements. Similar to beforementioned works, it does not address the issue with retrieving the context from the particular user.

3 Promising Solution

The notion Internet of Things is currently getting a lot of attention and the first real deployments are taking place in real-world scenarios. For instance, Gartner Inc. [3] predicts that by 2020 there will be 26 billion units installed in IoT products. Those devices can provide tremendous amount of information about the user's context. Especially the ones called "wearables". Nevertheless, even other forms of personal IoT devices, like smart homes, could provide us with plentiful of useful and valuable information.

Phone with GPS can provide precious location of its owner. Smart watches can do the same plus they can provide, for example, user's body temperature and pulse. First step of using those biometric information is to use them to form some kind of user's signature. For example, consider a car that would could measure weight and height of the owner. If someone with different body proportions would try to start the car, the car would require additional credentials (e.g. password entered through the entertainment system). This context-aware security system would solve the issue with passive keyless entry or keyless start that are vulnerable for theft [2].

Nevertheless, we can also use additional context data to alter security rules of the system. If we could measure blood pressure and pulse, we could guess the user moods e.g. stressed, angry, etc., and adjust the security of the system corresponding to it. Consider a very critical system, like stock trading or internet banking, if system would determine the user is nervous during performing the transaction with significant and unusual amount of money, it could ask for additional approval. For example, it could ask approval from a second trader or two-phase authorization to prevent wrong decisions based on actual emotions.

We focus on user context that is created based on near by devices to the user. This context helps the system to decide whether it should require additional approval or not. The reason is that most applications signs in or verify the user for the first time and then the session is maintained. An example is the OAuth protocol, when a token is created and assigned to the user. The token has a specific expiration time and when it expires then a new token is created based on the refresh token. However, the user is not asked to log in again. These approaches come with significant issues, for example: "How to decide whether the token was stolen?" or "How to decide whether this is really the user who was authorized in the first place?". The system could open sign in dialog and ask for username and password again or the system should use the two-factor verification (explained in case study), but this process should be initialized based on clues that alert the system. These clues might be user interaction with the system, or device that is used

or others devices that might be not directly involved in the interaction process between the device and software. The combination between the user interaction and IoT devices, that represent the indirect devices, are great choice for this type of situation.

We may observe IoT from several perspectives. We can focus on device itself or we can monitor the users, because every person has a specific set of behaviors and most of them has predictable time schedule. For example, the Google is able to decide where you work, where you park your car, etc. Based on that Google provides you morning traffic information and travel time estimation to the job. Your secretary knows when you usually come to work and what is your preferable restaurant, as well as she knows which car you use and what are your favorite hobbies, moreover she recognizes some of your friends. This implies the following: If somebody asks your secretary what are you doing in concrete time then she is able to predict what you are actually doing, because she knows you. In this section, we present a technique that helps machines to know you and based on your habits determine whether you should do additional verification when you want to use a specific part of software.

Unlike secretaries the IoT is not a human being and it does not pose prediction logic, but on the other hand it has access to sensors and devices on different places at same time. Your computer is connected to network via cable or Wi-Fi, therefore there is a specific device near to your location. The same applies for smart watch, fit bracer or another wearables device. When you are in a car then your mobile phone is connected with car via Bluetooth. Given the nature of the IoT, we can even use information from devices that are not connected directly to the user, but to one of his primary devices. We consider only devices that are connected with each other and we do not consider unconnected devices, because it is out of scope this paper.

Our solution represents connected devices as undirected graph. The edges connect two devices that interact with each other. There exist specific graphs for different situations. This means that the graph for office is different from a graph when the user is at home or when she or he moves from office to home in a car. The graph is also different

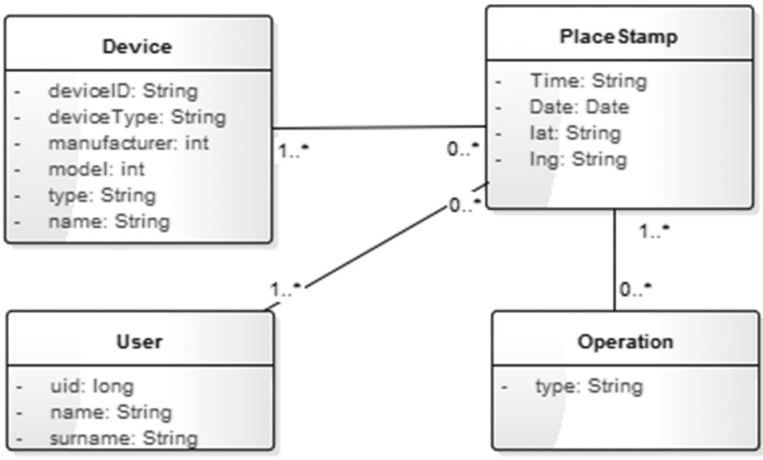

Fig. 1. Data structure to hold nearby devices

when the user practices any type of sport. The graph is created for the specific time and place, therefore it contains devices that the user usually uses in concrete time and place. Besides of graph we need meta-information of the graph. We created basic data structure that holds these types of information. It is represented in the Fig. 1. The data structure holds information about the device itself, time with place where it is used and the person who uses it.

The principle of this solution is to store data anytime when the user uses application. The user sends information about device that is used and he/she also sends information about other devices which are connected like smart watch, car Bluetooth or Wi-Fi. When the system has these types of information, then it is able to decide whether it is user's regular environment or it is not. The system stores signification amount of data, therefore it is critical to be aware of the time frame and aggregate data based on it. The basic aggregation and usage of this solution is demonstrated in a case study section; the detail information about how to implement this aggregation and how to make decision about user's confidentiality is matter of implementation and it should base on security rules in concrete usage.

4 Case Study

The proposed solution is demonstrated in the case study. We have chosen bank environment. The bank clients usually use internet banking. The software's task is to manage bank accounts. The user can work with transaction history, accounts, credit cards, loans, mortgages etc. For example, the user can change name of the accounts, create standing order, make request to offer, create payment, etc. The supported functionalities are different in each bank institution. However, most banks have one common functionality. It is two-factor verification when the user wants to create a payment. The payment could be created only by authorized user who is logged into internet banking. There exists a lot of login method for example: the user uses certificate with password or she/he uses login and password or combination with login, password and SMS authentication. We will consider only authenticated users and we will focus on the process of creation of a payment.

The process itself involves a lot of actions and preconditions. The user must be logged into internet banking, choose the source account, know the destination account, enter an amount and other details and finally, confirm the payment. There are a lot of processes that are triggered after the payment confirmation. The bank system has to verify whether it allows the user transfer the given amount from the source bank account, whether it is normal or suspicions operation and needs to authenticate the requesting user. The SMS two-factor verification is used to check the user's identity. This approach has various disadvantages. The user's identity and mobile phone could be stolen, unreachable, broken, or the provider is unreachable.

We simulated the data that could be obtained during the process in the Table 1. We store data from computer. The computer is connected via cable or Wi-Fi to Internet and it is also connected with mobile phone via Bluetooth.

Table 1. Examples of harvested data

DeviceID	Time	Lat	Lng	Operation	Source	User
Computer1	15:30:29	49.224	16.577	Login	Yes	mtomasek
Router1	15:30:29	49.224	16.577	NONE	No	mtomasek
Phone1	15:30:29	49.224	16.577	NONE	No	mtomasek
Computer1	15:32:15	49.224	16.577	Payment create	Yes	mtomasek
Router1	15:32:15	49.224	16.577	NONE	No	mtomasek
Phone1	15:32:15	49.224	16.577	NONE	No	mtomasek
Computer1	15:32:49	49.224	16.577	SMS verification	Yes	mtomasek
Router1	15:32:49	49.224	16.57	NONE	No	mtomasek
Phone1	15:32:49	49.224	16.577	NONE	No	mtomasek
Computer2	18:10:35	50.075	14.419	Login	Yes	mtrnka
Router2	18:10:35	50.075	14.419	NONE	No	mtrnka
Phone1	18:10:35	50.075	14.419	NONE	No	mtrnka
Computer1	15:25:58	49.228	16.577	Login	Yes	mtomasek
Router1	15:25:58	49.228	16.577	NONE	No	mtomasek
Phone1	15:25:58	49.228	16.577	NONE	No	mtomasek
Computer1	15:29:38	49.228	16.577	Payment create	Yes	mtomasek
Router1	15:29:38	49.228	16.577	NONE	No	mtomasek
Phone1	15:29:38	49.228	16.577	NONE	No	mtomasek

The table contains various information. It shows which device was used to access the network, the place where the user is and another device that she/he uses. The column Source indicates if the device is source of information that are in the table. The table is ordered based on date therefore, first twelve records are stored one day and the rest records are stored another day. The devices are the same when logging in, making the payment and verifying the payment. This is an initialization state and we require SMS verification in this state, because we do not know the user's behavior and environment. If the user creates another payment next day around 15:30 then we can compare connected devices with previous state in which was payment authorized. Moreover, we can compare place where the user is and if the place is the same, but the devices are different then the payment could be suspicious. We are able to create graph of devices that the user usually uses and their place in time. We can store any user who uses our application, therefore we can connect it together and compare their location in time to verify if their time schedule is usual.

The table represents another state. The second user (mtrnka) logged into internet banking at 6:10 PM. This is nothing special, however his computer is connected with the Phone1 that was used by another user mtomasek. If the phone is the authorization phone that belongs to mtomasek, then every payment creation by mtomasek should be suspicious. There is another case. It is the payment creation next day. We can compare connected devices, place and time and we can decide if the two-factor verification is

necessary or not. In this case, we might not to send the SMS on target device, because the phone is already somewhere around the computer, the user uses the same device, he is almost on the same position and he does the same action that he did yesterday in this time frame. We also could use these information as a fraud indicators and we can decide to use a different authorization method or to ban this transaction.

5 Conclusion

We presented an approach that targets the user rather than the system itself or actions in the system. Information from the user's nearby devices are used to obtain user context. The user's position, date, time and nearby devices by itself are critical parts of our method. The information is kept for future usage. When any decision about user behavior is needed, we can correlate current data with the historical data and tell whether the security rules should be altered. The basic data structure was presented in addition to the usage in an internet banking use case. We showed that our approach helps system to decide on the additional level of authorization necessity when the user's context is suspicious or unusual.

In future we would like to focus on the human health sensors. The sensor provides crucial data about the user, such as weight, hearth beat rhythm, etc. These data combined with nearby devices could provide more detailed information about user's context. Based on our approach we could decide more precisely whether the user exhibits some suspicious behavior. Integrating machine learning techniques in our decision scheme another direction we like to explore.

Acknowledgement. Research described in the paper was supported by the Grant Agency of the Czech Technical University in Prague, under grant No. SGS16/234/OHK3/3T/13 and by Technology Agency of the Czech Republic, under grant No. TH02010296.

References

1. Abowd, G.D., Dey, A.K., Brown, P.J., Davies, N., Smith, M., Steggles, P.: Towards a better understanding of context and context-awareness. In: Gellersen, H.-W. (ed.) HUC 1999. LNCS, vol. 1707, pp. 304–307. Springer, Heidelberg (1999). doi:10.1007/3-540-48157-5_29
2. Francillon, A., Danev, B., Capkun, S.: Relay attacks on passive keyless entry and start systems in modern cars. In: NDSS 2011 (2011)
3. Gartner Inc.: Hype Cycle for the Internet of Things (2013)
4. Harter, A., Hopper, A., Steggles, P., Ward, A., Webster, P.: The anatomy of a context-aware application. Wirel. Netw. **8**(2/3), 187–197 (2002)
5. Hong, J., Suh, E.-H., Kim, J., Kim, S.: Context-aware system for proactive personalized service based on context history. Expert Syst. Appl. **36**(4), 7448–7457 (2009)
6. Miroslav, M., Cerny, T., Slavik, P.: Context-sensitive, cross-platform user interface generation. J. Multimodal User Interfaces **8**(2), 217–229 (2014)
7. Ho, J., Intille, S.S.: Using context-aware computing to reduce the perceived burden of interruptions from mobile devices. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2005), pp. 909–918. ACM, New York (2005)

8. Kranz, M., Holleis, P., Schmidt, A.: Embedded interaction: interacting with the internet of things. IEEE Internet Comput. **14**(2), 46–53 (2010)
9. Petriu, E.M., Georganas, N.D., Petriu, D.C., Makrakis, D., Groza, V.Z.: Sensor-based information appliances. IEEE Instrum. Meas. Mag. **3**(4), 31–35 (2000)
10. Covington, M.J., Fogla, P., Zhan, Z., Ahamad, M.: A context-aware security architecture for emerging applications. In: Proceedings of the 18th Annual Computer Security Applications Conference, pp. 249–258 (2002)
11. Trnka, M., Cerny, T.: On security level usage in context-aware role-based access control. In: Proceedings of the 31st Annual ACM Symposium on Applied Computing (SAC 2016). ACM, New York (2016)
12. Farringdon, J., Moore, A.J., Tilbury, N., Church, J., Biemond, P.D.: Wearable sensor badge and sensor jacket for context awareness. In: Proceedings of the Third International Symposium on Wearable Computers, Digest of Papers, San Francisco, pp. 107–113 (1999)
13. Schilit, B., Adams, N., Want, R.: Context-aware computing applications. In: Proceedings of the 1994 First Workshop on Mobile Computing Systems and Applications (WMCSA 1994), pp. 85–90. IEEE (1994)