NEW APPROACH TO IOT AUTHORIZATION BASED ON SINGLE-POINT LOGIN AND LOCATION-SPECIFIC-RIGHTS

Tomasz Krokosz, Jarogniew Rykowski

Department of Information Technology,
Institute of Informatics and Quantitative Economics,
Poznań University of Economics and Business,
e-mails: tomasz.krokosz@ue.poznan.pl, rykowski@kti.ue.poznan.pl,
Poland

Abstract: The paper covers essential issues related to the protection of data packets transmitted among devices in Internet-of-Things networks. The basic idea is to use an approach in which overall security is maintained at the network-entrance level to free Internet of Things devices from the obligation to perform cryptographic calculations and to maintain encryption keys. The second goal is to parameterize the access rights by a physical place of login. The paper proposes a diagram of a generic way to authenticate a new-coming client in an IoT server using the Virtual Private Network (VPN) approach. The authentication depends on the place and time of connection. A user, navigating among specific locations and IoT network subparts, may miss or gain certain rights, depending on the situation, access grants, roles, etc. The text also describes a prototype implementation covering server application, operating on a Raspberry Pi microcontroller. Mobile clients connect and exchange data with the server via a dedicated smartphone/desktop application (programmed in C#). The application access is secured by a VPN client operating on Raspberry Pi.

Key words: Internet of Things, VPN, Internet of Things security, user rights.

1. INTRODUCTION

An organization of the method of accessing network resources, as a part of the generic network infrastructure, may be solved in several ways, passwords, certificates, and a reserved pool of IP addresses included. Typically, successful completion of the network login process results in access to the network resources, i.e., servers, printers, or computers. An important issue of the initial-connection process is to ensure the protection of transmitted messages to/from the network. This should be achieved in such a way that the data (despite the possible eavesdropping) would not be useful for any intruder. In response to the growing need for security of the network traffic, robust cryptographic data protection for IoT devices becomes is a priority. However, as the IoT devices are usually characterized by very limited resources (concerning CPU, memory, battery time, etc.), in most cases implementing

these procedures in the devices is extremely difficult. Small IoT devices, such as, e.g., a temperature sensor or on/off switch based on a relay, cannot perform certain cryptographic operations. However, in most cases, such devices are connected via wired and traditional connections (such as I2C, 1-wire, UART etc.), thus anyway secured in a different manner. Taking into account this observation, one needs to secure only the access to the proxy nodes, being brokers among the clients, and the limited IoT devices connected. It is crucial to propose an effective way to ensure a high level of security in the context of the Internet of Things taking into account the above-mentioned facts.

The aim of the paper is to present a way to implement a secure, eavesdropping resistant communication channel to the local network of Internet of Things devices, taking into account users' access rights varying in place and time. The goal was achieved by using Raspberry Pi as the main driver of the local network node, the so-called concentrator, with a VPN server installed, and treating this node as a communication base for the incoming mobile clients. As for the latter, a dedicated application has been prepared, making it possible to control the state of virtual devices (e.g., a thermometer, windows or doors) in a secured manner, via VPN connection. The server application was programmed in Python, while the client application was prepared using C#.

The further organization of the paper is as follows. The second chapter presents the idea of the Internet of Things. There, ubiquitous systems operating according to the rules defined by Mark Weiser and some issues related to the connection of mobile devices with the Internet are described. Special attention is put to the newcomer problem and the way of modelling and accessing local resources, granting permissions dynamically in an ad-hoc manner, with variable time, place, communication channel, as well as a generic level of security and privacy.

The third chapter contains a detailed description of two main problems considered in the paper: effective mobile-user authentication, and a provision of a secure communication channel. Possible implementations of the secured ad-hoc communication channel as well as access to local (at-the-place) IoT resources are described and discussed towards security and efficiency.

The fourth chapter presents the idea of using a local private network for user authorization purposes.

The fifth chapter covers several implementation issues. This part presents the basic methods of data exchange among devices, as well as the cryptographic base for the proposal.

The sixth chapter presents a sample scenario of using the newly-proposed authorization method in a quasi-real application.

The penultimate chapter presents a comparison of our approach and similar proposals.

The last chapter is a summary of the publication, describing also some directions of future work related to subject indicated.

2. INTERNET OF THINGS AND AD-HOC INTERACTION

Ubiquitous computing [10] is a phenomenon related to the dynamic expansion of the network applications as well as the devices connected. It implements the concept of M2M (machine to machine) connections, enabling communication, collection, processing, and data exchange among devices, without the need to involve people in the process. Such a set of communicating devices may be treated as a coherent ecosystem created to accomplish certain specific tasks. IoT-related technologies form the basis of research on the vision of ambient intelligence, which should be understood as a case in which the immediate human environment will be observed as an advanced (computationally and via efficient communication) technology. The system will be aware of the presence, personality, and human needs, and will be able to respond intelligently to indices about desires expressed by, for example, gestures, speech, or simply a presence at a certain place. Such a system, with machines exchanging information to help people, is now called the Internet of Things (IoT) [3] and Services (IoS).

The term mobile ad-hoc network usually means a decentralized wireless network that allows data to be transmitted as part of dynamically changing connections among nodes of this network. Mobile stations simultaneously perform the function of terminals and routers. This means that all mobile devices can send some data on their own as well as participate in the process of transferring some other data to the final recipient. In this perspective, the ad-hoc network is characterized by the lack of the need to use additional infrastructure, and mobile stations do not have nor a specific location in space neither fixed functionality and access rights. With regard to the presented type of a network, in addition to the problems which are typical for ensuring the transmission (radio band, coverage, routing of packets, etc.), one observes a “problem of the newcomer”.

A newcomer is a user (precisely, a personal mobile device belonging to this user) suddenly appearing at a place as asking for some help. The system task is to allow the access for this user (with respect to several parameters including user’s role and access grants). In turn, the system must somehow identify the users and determine if they have any privilege in the local network. One of the possible ways to implement the above-described process is to locate an additional node of the network, to which newcomers’ requests would be redirected. Such a node would be able to authenticate for users connecting to the network and would serve as an entrance to the network resources. Based on the authentication data, the user would be authorized to use the IoT devices and services, locally available via the authorization node. The scheme of the above-described solution is presented in Figure 1. The communication between authenticated nodes and the authorization node is marked by blue arrows, while the internal traffic among the IoT devices is marked in black.

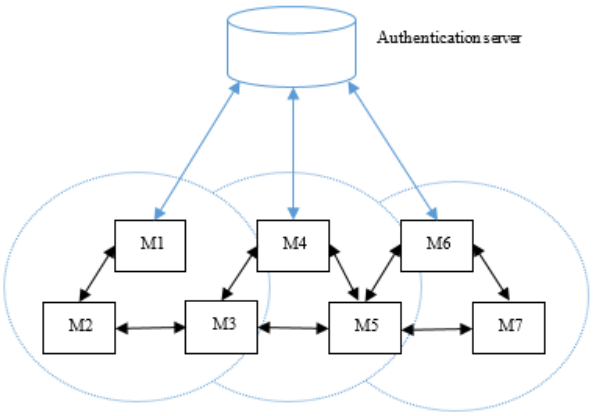

Fig. 1. Authentication in an ad hoc network

The technique of connecting a single node with a VPN network within an ad-hoc network (the blue lines at the figure) should be resolved similarly to a site-to-site VPN tunnel with an authentication server. This service allows network nodes (also connecting in an ad-hoc manner) to connect to another network via a public communication channel securely. This idea may be applied to the proposed architecture – the VPN would be used by a newcomer to authenticate, and the authentication node would also be used as a proxy to the local IoT devices and services.x.

3. THE PROBLEM OF AUTHENTICATING A NEWCOMER

As a part of the proposed solution, two techniques for solving specific problems should be considered and proposed, namely effective user authentication and secure transmission of data packets to/from local network the ad-hoc client is connected to.

The current user authentication standard is defined in the IEEE 802.1X standard [14]. It is implemented in most network devices as well as operating systems. It allows to conveniently and safely configure the network environment to be used by users. It also provides techniques for dynamic change of encryption keys used in network communication. 802.1X uses the EAP protocol - Extensible Authentication Protocol and includes multiple user authentication methods for both wireless and wired networks.

Cryptographic algorithms require hardware resources (memory, processor). Due to this fact, their use in the Internet of Things is not recommended. The issue of limited power sources is also important. The battery life of the device charged with the task of repeated cryptographic algorithms is effective shortened, which is unacceptable from the perspective of data exchange. For example, the use of cryptographic mechanisms in case of popular geolocation beacons (such as iBeacon), whose task is to periodically send a specific signal, shortens the device's working time from 4-5 years to several weeks only. Another consequence of creating a

cryptogram and generating a timestamp for the purpose of network transmission variability (resistance to "record and play" attacks) will generate a large amount of heat that will force the installation of a ventilation system and thus giving up the housing tightness and resistance certificate, e.g., IP68. The above requirements make it practically impossible to use encryption for the smallest network nodes – sensors and actuators.

The above-presented theoretical base for ad-hoc access of new-coming users may be extended to the idea of secured proxy access to the internal part of the local IoT network, i.e., IoT devices and their services. By means of the authentication proxy, controlling the state of IoT devices should be possible only for those users who have the appropriate permissions. Every unlogged (newcomer) user initially has no access to the IoT local network. To obtain some privileges, the user must confirm his identity by logging into the network. The fact of safe logging may be used to further verify proxy-based access to lower-level devices (unsecured network nodes), as described in the next chapter.

4. USE OF A LOCAL PRIVATE NETWORK FOR AUTHORIZATION PURPOSES

Devices connected to the network and the way of organizing data exchange with users create a local infrastructure, whose boundaries are determined by the local network. Both sensors and newcomers are connected to a shared local network, which is a natural filter for linking these two groups. Assuming that the sensors do not have direct access from the outside (i.e., they are not directly addressable outside their local network), at-the-entrance authorization at the network level is sufficient for the purpose of ensuring secure data transmission, form the point of view of newcomers. This observation may be justified by the arguments presented below.

Sensors, working with the use of industrial buses, e.g., 1-wire [16], I2C [17], SPI [18], etc., are usually connected by cables and wired links, which (1) enables power to be supplied with the same carrier, and (2) defines their physical protection. Such protection, in connection with the control of physical access to the locations, ensures, among others, confidentiality, integrity, as well as the availability of data and services. Most sensors devices are simple electronic devices that cannot be directly connected to an addressable network (that is, the nodes may be distinguished based on a unique name / address). As a rule, such a connection takes place through a more complex node (the so-called proxy). The above observation shows that the use of protections at the level of such sensors is practically impossible and not valuable. Instead, one should protect the whole network and proxy nodes, thus shifting the authorization from separate nodes to the entire system.

In this case, there is a need to authorize the first access of each ad-hoc connected device and to determine the roles (set of rights) of the users as they join the system. The network-entry protection eliminates the obligation of independent implementation of cryptographic procedures for all devices connected to the network

because they are sufficiently protected by the methods described above. Instead, one must enable strong authentication for the newcomers, which will determine their permissions for the entire period of contact with the network. The permissions will depend not only on the users themselves but also on the connection context, in particular - the place and time.

The area in which the control devices and the server were installed is usually geographically limited to a single building. As a consequence, this unit (a building or its part: an office, corridor, floor, etc.) was applied as the low-level system granularity. We assume that in each "location" there is a separate entry-point server capable of establishing VPN connections. As discussed earlier, enabling users to access the network and its resources must be preceded by the process of confirming users' identities and access rights. For this purpose, one must perform certain authentication and authorization operations. During the authentication process, the identity is verified as assigned to the persons (or workstations, data sources, devices, etc.). Once succeeded with the verification, the network acquires the belief that a given user / device / system is not dangerous to the network resources and thus is applied to use all (or a part) of them. After successful authentication, the authorization is performed, which provides some knowledge of whether a requested operation (or collection thereof) is allowed for a given user.

In the proposed solution, we applied the idea of a smart connection of a proxy and authentication servers, working for certain geo-location and serve all the local IoT devices. To simplify the description, we assume that each location is separately accessed (no overlapping with shared IoT devices), and for each of them, a private local network is created (Figure 2). In each of the separated subnetworks, the logged-in user may have different rights in relation to the IoT devices/services and possible actions to be performed. Each location has a separate, local VPN server, sharing authentication data with all the other locations (central authorization point, as described earlier). As a consequence, the problem of ensuring an adequate level of security has been transferred and imposed on these local verification points. In turn, the local Internet of Things devices (sensors and actuators) working in each of the local networks are exempt from the need to implement cryptographic algorithms.

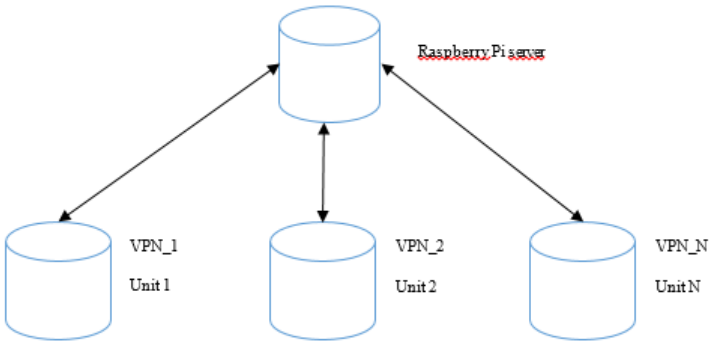

Fig. 2. Architecture and data flow for secure traffic to/from newcomer devices

Each of the created networks corresponds to a specific location or unit of a building (e.g., a room or their set with similar entitlements - for example, classes at a school). In each of them, logged-in users may have different rights regarding the devices and the operations to be performed in a given place and time. For example, a teacher is able to control his/her own office fully. However, starting the classes, the same teacher at a different location is able to control only the basic installations (light, projector, heating). The pupils are able to switch on the light, but only at the "public" locations), etc. As the users' devices migrate from place to place, the access rights are evolving, to be checked at each location by an independent network-entry point. Note, however, that the authentication process via VPN is the same for the whole network.

5. IMPLEMENTATION

To illustrate the idea, we created and used groups of several physical and virtual devices (sensors and actuators - thermometer, window and door controllers, etc.), accessible via the client (C#, Windows Forms), and server applications (Python, Raspberry Pi). We group the devices into (virtual) locations, establishing a set of possible commands and states (such as "a light may be switched on and off"). We also prepare an application to act as a support for local administrator, to manage the devices, roles, access rights, etc.

The client application is able to control the state of local IoT devices (with respect to the access rights, however) and to connect via a VPN channel to the nearest entry-point. We used popular OpenVPN library with 2K encryption keys. The entry points redirect the local requests to (1) authentication server and (2) local IoT devices. We somehow standardize and virtualize access to the devices [1] to enable easy access to the system regardless of time/place of such access.

For each of the device groups defined, we validate the possible states at the server-side. A client application, from which it is possible to manage devices without a physical contact, exchanges data with the server application using the https protocol. The server application recognizes the device the request came from, verifies the access rights, and then updates the information in the database, sending the appropriate command to the IoT devices. The exchange of packages between the client application and the server is performed within a VPN channel. The system architecture is presented in Figure 3.

It should be noted that the client device never communicates directly with IoT devices. Data exchange using VPN takes place only between the client and a trusted node (Raspberry Pi), which also acts as a proxy. So, protecting access to the proxy via a VPN connection de facto means protecting the entire network against unauthorized external access. As anyway the access to the address-free sensors and actuators may be implemented only via a proxy, this solution does not limit the system functionality.

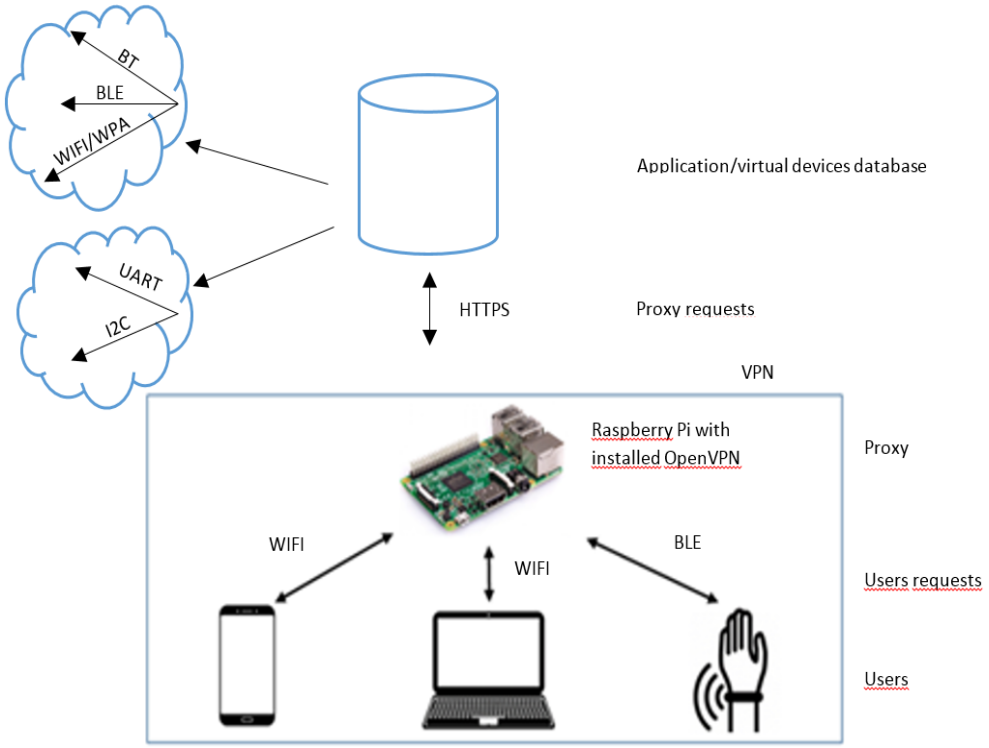

Fig. 3. Diagram of device-to-network traffic in a subnet /location

All entry points are logically connected to the same VPN (a VPN server has been installed on Raspberry Pi). The client sends a device state change request (or reading its status), and communicates with the server running on Raspberry Pi. The latter is able to recognize which device groups the request concerns, and to communicate using the https protocol with the virtual-devices database. After receiving the response, the application server sends a message to the client who started the transmission.

For each logged-in user, it is possible to collect their roles and access rights. This allows specifying whether a logged-in person who wants to make a certain action on a given device (e.g., to switch on/off the light) has the right to do it. Single-point login to the network allows accessing all network resources (in the scope of the entire building, even after changing the location and switching between subnets). The set of the currently available resources depends on the current user's location.

Each user may have different needs in relation to, for example, a level of location luminance or temperature. In this situation, we introduced an ability to select a level of luminance / temperature by each user, and then extracting the average value to be used as a final parameter for the device. The described use case is the basis of the democratic Internet [2]. Each user decides about preferred value, and the final setting is the quotient of the sum of values and their number (de facto, the arithmetic

average). Surely, the problem of ideal democracy may be fluctuated by some priorities, such as a teacher would probably have a greater priority to set the room temperature in comparison with the pupils. However, an ill pupil asking for a higher temperature would increase the priority, etc. In general, the problem of democracy on the Internet of Things is much more complicated than this simple example, but the discussion on this subject definitely goes beyond the scope of this article.

Regarding the access rights assigned to users, there may be a case in which a company employee has access to specific areas of the building, and in certain rooms, the ability to change the status of devices. Moving from point A to point B within the building, a user switches among separated VPN networks, and, in each network, the final permissions may not be the same (or they are, but some others also apply). In one unit a user can, for example, open all doors freely and change the status of all devices (his workroom), while in other locations the permissions may be limited to for example temperature changes, but no longer controlling the state of the window or opening the door (e.g., a meeting room). In general, access rights to devices and rooms and the ability to change the status of IoT devices depends not only on the mobile user, but also on the location, and even time (e.g., opening a window at night is forbidden).

6. USE CASE

One of the applications implemented as part of the prototype is the server application (written in Python), which was launched on Raspberry Pi. The mentioned microcontroller listens whether any device authorized to be activated on the network sends a request, for example, to read or write data. This information is used to control the client interface – each client is able to see only the functionality that is provided at a given place/time and for this person. Thus, the application interface may vary as the user migrates from place to place.

Assigned rights define what processes the user can perform and with which devices. If the user tries to change certain device status, the request is detected by the proxy. There, the request is supplemented with the local sensor data, as well as the target (reachable, desired by the user) state. At the moment when the users change their locations, they are switched to other networks (e.g., other branches of the company). The same request at a different location, unchanged from the point of view of a user, will be, however, re-mapped to another local IoT device. Note that access to different devices and locations may vary according to the location – the interface is automatically adjusted to vanishing and new at-the-place functionality. It means that access rights are evolving (as a consequence, the application interface will also be changed), and some actions that were possible to execute at a certain sub-network (such as closing a window) at another location will be impossible to achieve.

In everyday practice, it may look as follows. A clerk working in his room joins the node as the "owner" of this room. He has full rights to use all his local devices,

such as lights, air-conditioning, personal PC, etc. However, if he moves to the next room to visit a friend, his rights at the new place will be limited to the light controller. After returning, he will regain his "owner" rights. The control interface on his smartphone will dynamically include, remove, or add appropriate controls. It should be clearly stated that in both cases, the authorization data are presented and processed in exactly the same way in relation to the local network, and the whole transmission, as performed over a VPN channel, is always protected against unauthorized access.

The main window of the client application is shown in Figure 4. Note that the presented figure is certainly not a target interface of the application. It should be rather treated as an illustration of the auto-adaptation of an interface and auto-filtering the set of devices and actions available at a given location.

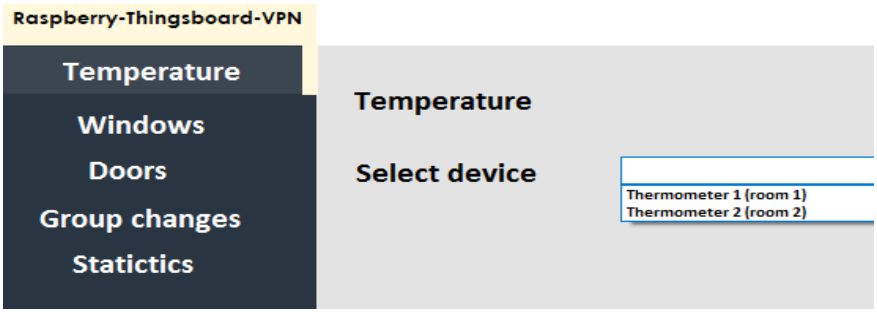

Fig. 4. Main window of the client application at certain location

In its left part, there is a menu section from which the user defines the target group of devices. Clicking on a button with a specific label causes refresh the view in the middle of the window, which will initially contain a list with all the devices available in the system of the selected type. Changing the position on the list causes the view under the list to be refreshed and selected the data about the device (Fig. 5).

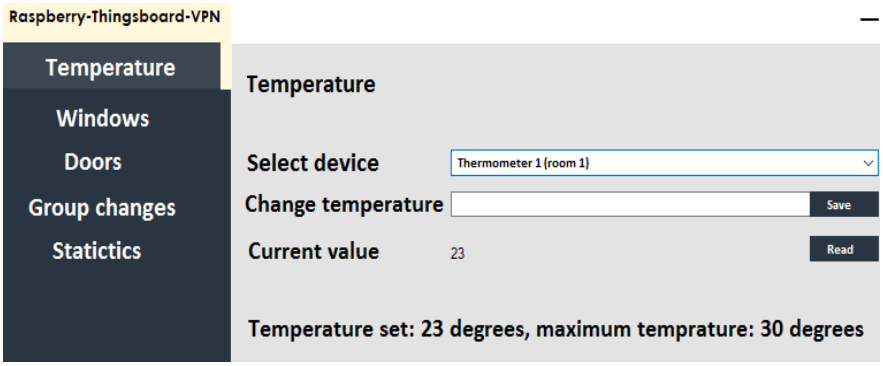

Fig. 5. Temperature section of the user interface at certain location

From this level, the user can read the current value as well as change it. Each action (query) is done using a proxy (Raspberry Pi), on which the server application was launched (programmed in Python, Figure 6).

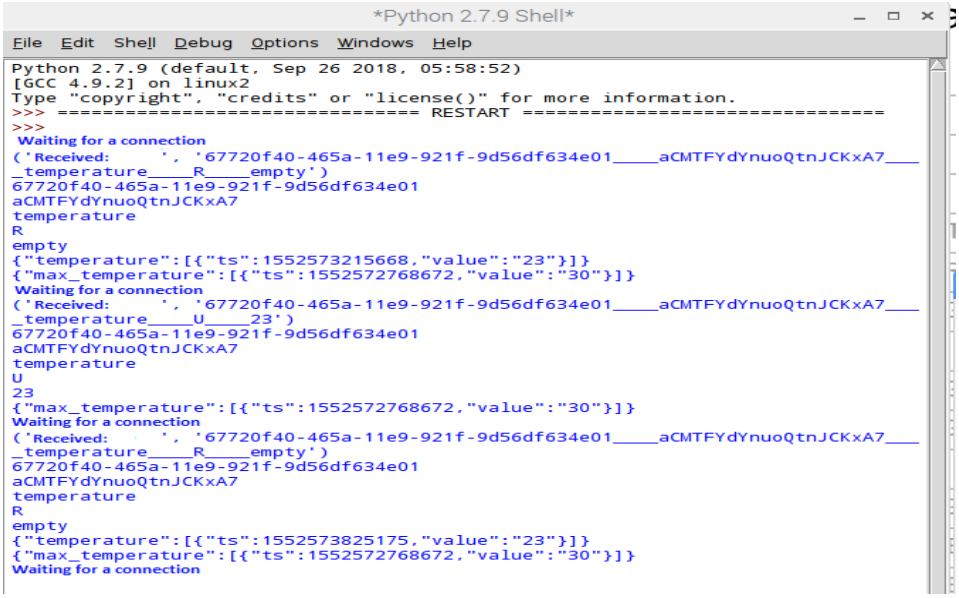

Fig. 6. Server application (Python log)

A recognized request from a client starts the process of communication with the database, which contains a complete set of information on all device groups and their values (Figure 7).

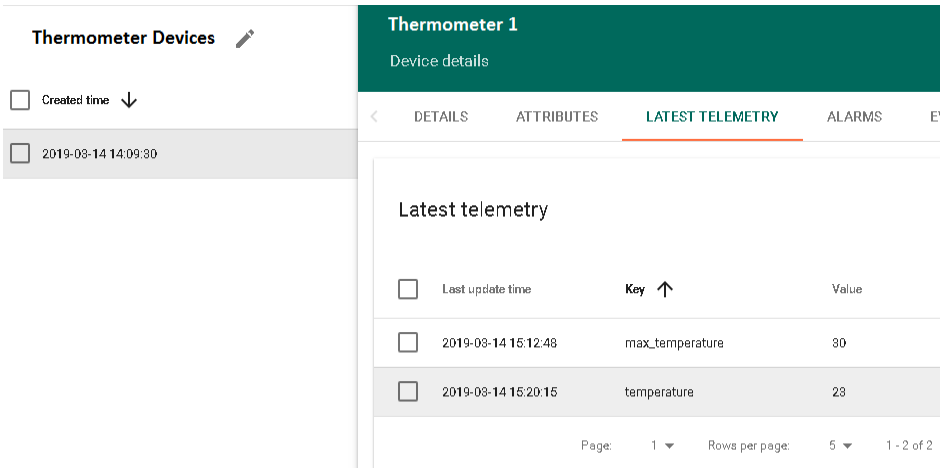

Fig. 7. Group of virtual devices – administrator view to the database

In case of a change of a location (e.g., after visiting a colleague in another department of the company), the local sub-network the user is connected to is also changed. As a consequence of this switching, an automatic change of the user interface is observed (Figure 8), which is a response to location change and updating the access rights in a given section of the building.

Fig. 8. Main window of the client application after changing location – evolved set of devices available

In this case, as a consequence of the change of location, access to a single thermometer is possible only. In addition, the available device categories are restricted (there is no access to the window control module in this subnet).

The presented case is an example of using the application to change the temperature and to read its value. The section of windows and doors is very much like the one currently presented. From this level, it becomes possible to manage and read the state of objects. It is possible to manage a group of devices rather than addressing each one separately at a time - with a single click, one can change the status of all devices assigned by current access rights (i.e., to close all the windows at once, not one by one). The last menu position - statistics module - provides information on the number of shares and their types, as well as the person who was responsible for changing the status (used mainly for validation).

7. COMPARISON WITH SIMILAR WORK

As a result of the growing need to protect data in the Internet of Things environment, several papers have been published with attention paid to safety/security/privacy. However, to our best knowledge, only a few point VPN as a primary authentication technique – some of them are briefly discussed below and compared with our approach.

In the article [7], the authors adopted the assumption that a separate VPN network will be created for each device group, which results in meeting the challenge related to the scalability of the solution. Each application has a dedicated VPN overlay, therefore used sets of services are separated. However, it is not known how devices behave when they are connected to another network, i.e., whether they will have previous permissions or those that they have due to their location (i.e., the network they are currently connected to). In our solution, the authentication process is assigned to the network, but the access rights are directly dependent on the location.

The article [6] contains scenarios of analysis of security and privacy threats for typical smart home architecture with limited processing possibilities. The authors focused on security errors that designers may commit while creating the Internet of Things environment. In our solution, we protect the entrance to the network, and each node without prior authorization has no access. Despite noticed by the authors

growing capabilities of Internet of Things devices, our solution allows removing the obligation to perform cryptographic operations from sensors while ensuring transmission security. As a result, the device's operating time is undeniably longer.

In [12] the author used Raspberry Pi working as a compute node (using OpenStack) and being a part of cloud-based solution, for a smart-city application. Our proposal was to apply IoT at home, mainly for non-commercial applications. Thus we deal with different problems, such as varying ways of energy supplementation (usually cable-based, local connections) and related cost/computational power restrictions, while these were not taken into consideration for Luchian's work.

In [11], the author proposed the usage of VPN and Wireless Sensor Network to create a virtual environment. With VPN, each IoT device gains its own virtual environment, which enables using a dynamic IP address to connect to other IoT devices and their virtual environments. For this solution, thanks to using WSN, virtual environments are able to identify any data that stand for potential danger. This solution is quite similar to the earlier-described proposal [7] and does not take into consideration the mutual relationship between the access rights and the location of a request.

Author of another proposal [13] notes the threats and security restrictions resulting from limited resources of the IoT devices. The primary purpose of this paper is to improve the throughput of the application level of the IoT gateway while conforming to the security and privacy guarantees simultaneously. In this solution, a flow-level adaptive mobile VPN solution is proposed tailored for IoT ecosystems. Again, this proposal does not take into consideration the location of a request.

Article [5] is a general overview of problems related to the security of entities existing on the Internet of Things. Researches focused on security gaps resulting from data exchange technologies within the Internet of Things devices. In our solution, the newcomer must successfully pass the authorization and authentication process to have access to resources. Securing the entry to the network, which has been described in previous chapters, guarantees access only for authorized users, and protecting only one entry point to the network is definitely simpler and more effective than detecting and fixing security gaps for access to multiple devices. Exchanged data packets among devices, even if they are intercepted, will be in an encrypted form, which the intruder will not be able to use in any way.

The authors of the article [8] reviewed the progress of research on the Internet of Things in relation to maintaining security. They listed key groups of issues, including encryption, communication security, data protection as well as cryptographic algorithms. However, the implementation of any cryptographic method (even if it is a comparatively low computational TEA encryption, let alone about AES [15]), will effectively reduce the device's working time, in addition, cryptography of devices connected by local industrial buses (I2C, 1 wire, SPI) is unnecessary in the context of network node (proxy) protection.

The issue of ensuring the security that we are dealing with is also described in the article [4]. The author indicates the threads occurring in the transport layer and the application layer, and provides possible ways to prevent them. Counts to them user authorization, a set of possible protocols, a firewall as well as the use of cryptographic algorithms. The mechanisms proposed by the author will successfully provide security for data transmission, but effectively increase the energy demand of devices.

The authors of the next article [9] discuss various applications of the Internet of Things and possible threats. Similarly to [4], authors listed individual layers and potential threats associated with them and exchanged threats to privacy and security at various levels. It is a general study, which does not offer any detailed solutions, in particular, it does not differentiate access rights depending on user location.

Mentioned papers are related to security issues, which with the increasing use of the Internet of Things requires extraordinary measures. They are not always possible to implement (including hardware requirements); therefore, there is a need for applying a different organization and system architecture scheme. In addition, the papers do not raise the issue of user rights depending on the location, so only our proposal is complete.

8. FINAL CONCLUSIONS AND DIRECTIONS FOR A FUTURE WORK

The text describes the protection of data sent among IoT devices and mobile clients, ad-hoc connecting to an IoT sub-network. Security of both the transmission and the devices is ensured, among others, by applying authorization at the network level. The new idea is that each user, while connecting to a local sub-network using a VPN client, is granted access rights, which are assigned at the time of joining the system and then extended to the IoT subnetwork(s). Surely, the use of a VPN server allows for secure data transmission. However, what is new for our proposal, it removes the obligation to encrypt data by Internet of Things devices directly. The main advantages of the proposed solution are related to enabling encrypted data transmission while not reducing the life span of sensors and actuators. The mentioned devices have small computing power and limited power sources. Therefore, there was a need to transfer security assurance to another level and release devices from the obligation to implement cryptographic data protection algorithms. In addition, the method used to assign grants to users depends on the location sub-network they are currently connected. This is a new and convenient way to manage access rights and gateways (network entry points) at the same time, which may be used at many locations, with a hierarchical and well-defined population of users (such as the employees of a company). Employee rights change dynamically and depend on the place where they are currently located, and only those users who successfully complete the authorization process at given place have access to the particular, local part of the network.

This paper does not provide a solution to the problem of identifying the place and time from which the user gains access. For this purpose in the test implementation, we used QR codes – reading the code assigned to a given service (place), the user sends an access request, appropriately addressing and encrypting the communication, and then accessing the service interface. A possible sample replacement for QR codes is associated with the use of Near-Field-Communication NFC tags, or Bluetooth Low Energy (BLE) marketing channel, BLE Mesh and autonomous BLE devices such as geolocation beacons. However, choosing the appropriate geo-location technique is a purely technical issue that goes beyond the scope of this article. We used QR-codes for the first implementation as this is a pretty straightforward and easy-to-implement technique, but we now moving to BLE and BLE Mesh, and proposed a so-called BLE model for better solving the "newcomer" problem (described earlier in the text) in a more secured way. Anyway, this choice does not change the idea that is described in this paper. Thus we do not provide the details regarding this topic, which in turn is quite complicated and needs a separate publication.

The technical requirements for devices performing cryptographic tasks are also not described (including smartphones as basic access devices for people), similar to the back-end level of the authorization environment - databases, WWW servers, and Radius, etc. The implementation of the system's prototype is still at an early stage; therefore, at the moment, we cannot provide evaluation data or describe solution tests.

Currently, we focus on the aspect of protection of transmitted data packets and the above-mentioned secured and trusted geo-tagging using BLE Mesh, as briefly introduced across the text. In the longer term, we will start a more generic, separate work related to the issue of the efficient site- and time-based identification for ad-hoc applications.

REFERENCES

[1] XXXX removed for anonymization purposes

[2] Wójtowicz A. Architecture for adaptable smart spaces oriented on user privacy, *Logic Journal of the IGPL*, ISSN 1367-0751, e-ISSN 1368-9894, 2017.

[3] Ashton, K. That 'Internet of Things' Thing, *RFiD Journal*. 22. 2009, pp. 97-114

[4] Gupta, J. Security and Privacy Issues in Internet of Things (IoT). *International Journal of Research in Computer Science*, 2, 2015, pp.18-22

[5] Geneiatakis, D. Security and privacy issues for an IoT based smart home, *40th International Convention on Information and Communication Technology, Electronics and Microelectronics* (MIPRO), 2017.

[6] Shif, L. Improvement of security and scalability for IoT network using SD-VPN, *2018 IEEE/IFIP Network Operations and Management Symposium*, NOMS 2018, Taipei, Taiwan, 23-27 April 2018; Category numberCFP18NOM-ART; Code 137784.

[7] Kulkarni, S. Internet of Things (IoT) security, *Proceedings of the 3rd International Conference on Computing for Sustainable Global Development*, INDIACom 2016, 27 October 2016, Article number 7724379, pp. 821-824.

[8] Husamuddin, M. Internet of Things: A study on security and privacy threats, *2nd International Conference on Anti-Cyber Crimes*, 19 April 2017, Article 7905270, pp.93-97.

[9] Heyman, K. A new virtual private network for today's mobile world, *Trade Journal*, ISSN: 00189162, DOI: 10.1109/MC.2007.410

[10] Weiser, M. Some computer science issues in ubiquitous computing, „*CACM*", vol. 36, issue 7, 1993, http://dx.doi.org/10.1145/159544.159617.

[11] Majumdar, P. Combination of Virtual Private Network and Wireless Sensor Network: Protection Against the Interference Problem of IOT, *8th International Scientific Conference on Engineering, Technologies and Systems*, TechSys 2019; Plovdiv; Bulgaria; 16 May 2019

[12] Luchian, E. F., Taut, A., Ivanciu, I. A., Lazar, G., Dobrota, V. Mobile wireless sensor network gateway: A raspberry Pi implementation with a VPN backend to OpenStack, *25th International Conference on Software, Telecommunications and Computer Networks*, SoftCOM 2017.

[13] Hussain, S. R. Securing the insecure link of internet-of-things using next-generation smart gateways, *15th Annual International Conference on Distributed Computing in Sensor Systems*, DCOSS 2019.

[14] Configuring IEEE 802.1X Port-Based Authentication, https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst6500/ios/12-2SX/configuration/guide/book/dot1x.pdf

[15] AES, Advanced encryption standard, https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.197.pdf

[16] 1-wire, Overview of 1-Wire Technology and Its Use, https://www.maximintegrated.com/en/app-notes/index.mvp/id/1796

[17] I2C, I2C-bus specification and user manual, https://www.nxp.com/docs/en/user-guide/UM10204.pdf

[18] SPI, Serial Peripheral Interface (SPI) – User Guide, http://www.ti.com/lit/ug/sprugp2a/sprugp2a.pdf.

Information about the authors:

Tomasz Krokosz, M. Sc, Ph.D. Student, Department of Information Technology, Institute of Informatics and Quantitative Economics /Poznań University of Economics and Business. Research interests: Programming and programming languages, Cryptography, Database design and programming, Internet of Things.

Jarogniew Rykowski, Ph.D. Associate Professor, Department of Information Technology, Institute of Informatics and Quantitative Economics – Poznań University of Economics and Business. Research interests: Internet of Things, Distributed Systems, Calm computing, Ad-hoc and multi-hop networking, including telematics applications.

Manuscript received on 9 December 2019