Industrial Engineering and Applications
L.-C. Tang (Ed.)

doi:10.3233/ATDE230071

Utilizing Key Internet of Things Concepts in Improving the Performance of Card-Based Access Control Systems for Vehicles in Truck-Loading Fuels Terminals

Moatz M. Bahgat [a,1], Hania H. Farag [a] and Onsy Abdel Alim [a]
[a] *Faculty of Engineering, Alexandria University*

Abstract. The Internet of Things (IoT) is viewed as the umbrella under which heterogeneous devices are connected to form what is called the network of the future (NoF). In the IoT era, all objects and devices are instrumented, interconnected, and interacted with each other in a smart manner. Those smart Things gather huge amounts of data from the real world and stream them up to the digital services running in the cloud. This paper; however, attempts to demonstrate practically that the Internet of Things is more than just connecting the cyber and physical worlds, but it may also improve the performance of existing systems and applications drastically if its concepts were used efficiently. Many different types of communication systems and networks might benefit from the efficient utilization of IoT concepts, being architectural concepts, communication models, or key IoT-enabling technologies.

Keywords. IoT, RFID, SOA, EDA, Web, Client, Server, HTTP, Access Control, Embedded, Gate

1. Introduction

According to the International Telecommunications Union (ITU), the Internet of Things (IoT) is defined as a global infrastructure for the information society, enabling advanced services by interconnecting physical and virtual things based on existing and evolving interoperable information and communication technologies [1]. Through the exploitation of identification, data capture, processing and communication capabilities, the IoT makes full use of things to offer services to all kinds of applications [1], [2], [3]. That is, the Internet of Things (IoT) describes the network of physical objects i.e. Things, which are embedded with sensors, computing capabilities, software and other technologies for the purpose of connecting and exchanging of data with other devices and systems over the Internet or other similar communication networks [3], [4], [5]. In the IoT era, all objects and devices are instrumented, interconnected, and interacted with each other intelligently

[1]Corresponding Author: Moatz M. Bahgat, e-mail: moatz.bahgat@gmail.com.

with the IoT aiming to link the real world to the digital world i.e. Cyber-Physical communication and connectivity [6], [7], [8].

However, IoT is more than just connecting Things to the Internet. This paper highlights the notable IoT concepts which, if utilized efficiently, would enhance the performance of existing systems and applications drastically as shall be discussed next.

2. Notable IoT Concepts

This section sheds some light on a range of the notable concepts of IoT. There are IoT concepts related to the architecture of IoT systems, the communication model followed by IoT systems and networks, in addition to key IoT-enabling technologies which have led to the evolution of IoT.

2.1. Layered Service-Oriented Architecture

The IoT system has a layered architecture, adopting the service-oriented architecture (SOA) model [9]; however, there is no single consensus on the architecture of IoT. Different layered architectures with different notations have been proposed by different researchers [8] including three-, four- and five-layered IoT architectures [7], [8], [10]. For the sake of clarity, the three-layered IoT architecture shall be considered as it defines the main idea of the Internet of Things (IoT). As shown in Figure 1, the three layers are:

1. The Identification Layer,
2. The Network Layer and
3. The Service Layer

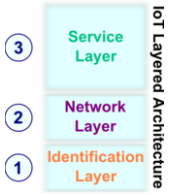

Figure 1. IoT Layered Architecture.

2.1.1. Identification Layer

Which is also called the Perception Layer, and in other literature the Sensing Layer [9], is the physical layer, which has sensors for sensing and gathering information about the environment. It senses some physical parameters or identifies other smart objects in the environment [2], [7], [8]. This layer is the origin of information and is considered to be the core layer of IoT, in which all kinds of information from the physical world used are collected and perceived [2], [3]. There exist many technologies which enable IoT. Key IoT enabling technologies are Radio Frequency Identification (RFID), Global Positioning System (GPS) and various sensor technologies including temperature, humidity, and Wireless Sensor Networks (WSNs) [9].

2.1.2. Network Layer

The Network Layer, which is also called the Transport Layer, is responsible for connecting to other smart things, network devices, and servers. Its features are also used for transmitting and processing sensor data [2], [8]. Network connectivity is considered the foundation for IoT, and the method of connectivity shall be based on the IoT application and the communication technologies used. Those technologies may be applied to the different types of IoT communication networks including wireless and wired connectivity [8].

Some of the well-known IoT networks are the ZigBee, Wireless Fidelity (Wi-Fi) & its low-power counterpart (Wi-Fi HaLow), Long Range Wide Area Network (LoRaWAN) and Ethernet networks [2], [6].

Communication protocols for IoT include HyperText Transfer Protocol (HTTP), Constrained Application Protocol (CoAP) and MQTT a client-server, broker-based publish/subscribe messaging protocol which are means for providing lightweight data transport [2], [4], [11], with several IoT protocols being published by the Organization for the Advancement of Structured Information Standards (OASIS) [8], [12].

2.1.3. Service Layer

This top layer, which is also known as the Application Layer, is formed by IoT-driven applications i.e. it is responsible for delivering application-specific services to the user based on the networked data. It defines various applications in which the Internet of Things can be deployed, for example, smart homes, smart cities, and smart health [8], [13]. This layer is mainly responsible for processing and managing of the data transferred [8]. It stores, analyzes, and processes huge amounts of data that comes from the identification layer through the network, providing a diverse set of services and applications. This layer employs many technologies, mainly web-based, including database management systems, cloud computing, big data analysis modules and business models [8], [10].

In order to support the large burst of data flow transferred by the large number of connected devices in the Internet of Things, the concept of Fog computing, which is also known as Edge computing, has emerged [14], [15]. Fog computing aims at extending the cloud computing by adding computational power, storage and communication capabilities to the edge of the network, in support of the IoT [15].

2.2. Client-Server Communication Model

In the distributed IoT system of heterogeneous sensors and devices, i.e. things, the things are connected to the cloud using the well-established client-server architecture [1], [9]. The sensors and devices at the Perception layer of the IoT system act as the clients which communicate with the server running in the IoT Service layer, i.e. the cloud, and possibly through an IoT gateway too [3], [16].

Traditional internetworking is based on the well-known Transmission Control Protocol / Internet Protocol (TCP/IP) protocol architecture which is a result of protocol research and development on the experimental ARPANET, funded by the Defense Advanced Research Projects Agency (DARPA) [12], [17].

In client-server TCP/IP-based communication protocols, the data is exchanged between the client and the server by means of a series of request-response messages in

which a client sends a request message to the server, which in turn replies back with a response message [9], [17] as shown in Figure 2.

Figure 2. Simplified IoT Client-Server Communication Model.

A well-known example is the HyperText Transfer Protocol (HTTP). HTTP is a stateless connection-oriented TCP/IP application layer protocol which requires a reliable network connection, with a valid TCP/IP port number, between the client(s) and the server for data exchange between them [12], [17]. Various IoT communication protocols build on HTTP e.g. CoAP which provides HTTP-like application protocol for constrained devices [4], [9].

2.3. Event-Driven Architecture

The IoT system follows an event-driven architecture (EDA) in which data transfer is triggered when a certain event occurs [9]. EDA is common in IoT applications built with microservices where communication between decoupled services is triggered by events [18]. Event-driven architectures usually have three key components [18], [19]:

1. Event Producers
2. Event Routers
3. Event Consumers

An event producer generates an event, which is in turn filtered and pushed to the event consumer through the event router [9], [18]. The event router establishes interoperability among the heterogeneous systems, so they can exchange messages and data while remaining agnostic [18]. Producers are decoupled from consumers and consumers are decoupled from each other, this facilitates services scaling and updating independently [9], [18], [19].

An event-driven architecture might use a publish/subscribe, i.e. pub/sub, model as in the case of MQTT or an event stream model [9], [19]. The components of the IoT event-driven architecture (EDA) are shown in Figure 3.

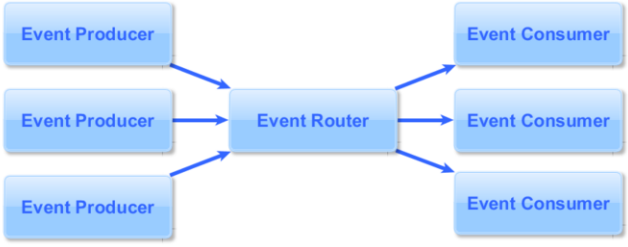

Figure 3. IoT Event-Driven Architecture.

In the next section, an overview on one of the communication systems which may benefit from the efficient utilization of the notable IoT concepts is discussed.

3. Traditional Card-Based Access Control Systems for Vehicles in Truck-Loading Fuels Terminals

An Access Control System (ACS) for vehicles in Truck-Loading Fuels Terminals is a communication system which is used to permit the entry or exit of authorized vehicles. There exists various access control systems which use different techniques, mechanisms and communications protocols; however, the most common method in controlling the terminal access is to use access cards [20]. A typical Card-based Access Control System for vehicles in Truck-Loading Fuels Terminals is shown in Figure 4 and consists of [21]:

1. Card Reader
2. Host Server
3. Communication link

Figure 4. A typical Card-based Access Control System for vehicles in Truck-Loading Fuels Terminals.

Typically, communication between the gate card reader and the host server takes place over a serial line including serial RS-232 and serial RS-485 for point-to-point or multi-drop access control systems respectively [22], following a master-slave communication model [22], [23]. Despite the fact that serial RS-485 cables provide distances up to 1200 m; however, they can only communicate with baud rates, i.e. serial data rates, up to a maximum of 1 Mbps, that is, low data rates when compared to Ethernet network cables which provide higher communication speeds [17], [24], [25].

Moreover, in many traditional Card-based Access Control System for vehicles in Truck-Loading Fuels Terminals, the host server keeps polling the access control card reader for data periodically, i.e. polling communication, wasting a lot of the server processing power and the available network resources which result in degraded system performance [17], [26].

Many Card-based Access Control System for vehicles in Truck-Loading Fuels Terminals, are still utilizing read-only cards which have several performance issues. An example of such cards are the Wiegand cards which operate according to the Wiegand effect [27], [28]. The most popular Wiegand Card format is the 26-bit Wiegand Card Format which is the standard Card Identifier length and is a widely used industry standard [29]. It consists of 24 code bits and 2 parity bits. The 24 code bits are divided to an 8-bit facility code and a 16-bit card identification number as shown in Figure 5.

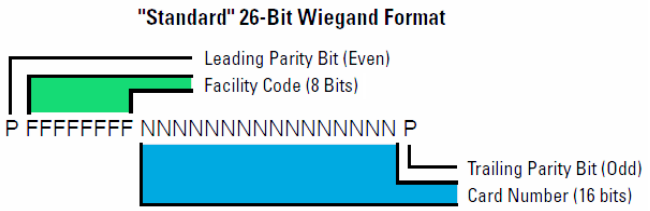

Figure 5. The standard 26-bit Wiegand Card Format.

The 26-bit Wiegand Card Format is an open format which is publicly available with no restrictions on its use or duplicating the card code numbers [29]. Although being used for decades, Wiegand cards have several limitations including:

- The physical size of the Wiegand card limits the maximum length of Wiegand cards identifiers to less than 37 bits
- Because the Wiegand card is a Read-Only access card, it is prone to the card cloning security attack

Compared to RFID, one of the most prominent IoT-enabling technology, RFID cards provide the following added features [2], [21]:

- Data writing capabilities
- Unique identification codes with longer length

Although they are commonly used in Truck-Loading Fuels Terminals, traditional Card-based Access Control Systems have several performance issues regarding the speed, the security and the utilized resources consumed during their operation.

4. Developed IoT-Based Online Access Control Systems for Vehicles in Truck-Loading Fuels Terminals

In this section, a developed IoT-based Online Access Control System for vehicles in Truck-Loading Fuels Terminals and the benefits of utilizing the IoT concepts shall be described in more detail.

4.1. System Architecture and Components

The developed system was designed to follow a client-server model in which the gates card readers acted as the clients which communicated with a web server. According to the 3-layer IoT system architecture, the developed system may be divided into three parts; Identification Layer, Network Layer and Service Layer, as shown in Figure 6.

4.1.1. Identification Layer

In the developed IoT-based Online Access Control System, the RFID technology was used at the Identification layer of the IoT layered architecture [3]. A RFID reader installed at the terminal gate detected a RFID card and read both its unique identification number (UID) and the data stored on it. Each entry or exit gate was equipped with a card reader embedded system which consisted mainly of an Atmel AVR 8-bit microcontroller,

Figure 6. Developed IoT-based Online Access Control System architecture.

an Ethernet module, a RFID proximity card reader module and a micro Secure Digital (microSD) card reader module [30].

The embedded system components used Serial Peripheral Interface (SPI) [23], [24], [25] for communication with the ATmega AVR microcontroller [31]. The block diagram for the gate RFID card reader is shown in Figure 7.

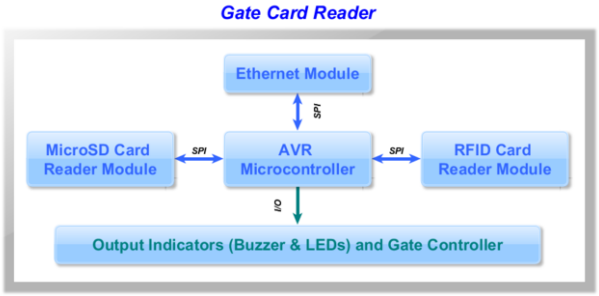

Figure 7. Block diagram of the gate card reader components.

The developed system was a Card-based Access Control System which utilized the RFID technology, one of the most prominent IoT technologies [9]. The RFID card reader and writer operated at 13.56 MHz and supported the ISO/IEC 14443 standard [2], [21]. The role of this module was to:

1. Scan for a compatible RFID access card,
2. Read the unique identification number (UID) and the data of the detected card.

4.1.2. Network Layer

In the Network layer of the IoT layered architecture, the information gathered at the Identification layer was transferred between the developed IP-based gate RFID card reader and a web server, running in the Service layer, through an Ethernet cable, which was connected to the gate card reader embedded Ethernet module, using HyperText Transfer Protocol (HTTP) version 1.1 i.e. HTTP/1.1 [16], [17]. Following the IoT communication model which is based on client-server communication, the gate card readers acted as the web clients which were communicating with the web server.

4.1.3. Service Layer

The Service layer of the IoT layered SOA architecture is the layer at which the data collected at the Identification layer and transmitted through the network is processed and managed [3], [4]. The Service layer may or may not be located in the same geographical location as that of the Truck-Loading Fuels Terminal. In case the Service layer and its components were located in a different location, then those services would be provided as a public cloud, whereas if located on the Truck-Loading Fuels Terminal premises it would be a private cloud [12], [32]. There were two main components which provided these services in the developed IoT-based Online Access Control System:

1. Database Management Server
2. Web Server

In comparison to traditional Card-based Access Control Systems for vehicles in Truck-Loading Fuels Terminals, and due to efficiently utilizing the notable IoT concepts, the new design approach followed in developing the IoT-based Online Access Control System offered enhanced performance as shall be demonstrated in the next section.

5. Practical Experiments & Results

5.1. Speed Performance

Regarding the communication between the web server and the gates card readers, and as result of building upon the IoT client-server model rather than the master-slave model, Ethernet cables were used rather than using Serial cables. Ethernet cables provide significant communication speed performance as compared to serial RS-232 or RS-485 cables [24], [33], [34]. This is shown in Table 1.

Table 1. Comparison between Serial RS-232, RS-485 and Ethernet communication cables and networks.

Point of Comparison	Serial RS-232	Serial RS-485	Ethernet
Maximum cable length (m)	~15	~1200	~100
Maximum data rate	115.2 kbps	1 Mbps	1000 Mbps
Maximum number of network nodes	1	32	1024

In addition, the developed IoT-based system followed an interrupt-driven design approach, i.e. event-driven architecture, in which the communication between a gate card reader and the web server only took place once a card was detected instead of repeated continuous communication i.e. polling communication. This reduced the amount of network traffic and thus no noticeable delays would occur, hence improving the system performance. Moreover, and due to utilizing the RFID technology, the reading of an access card took ~0.5 msec. as compared to ~55 msec. in case of Wiegand access card i.e. faster access card reading time. Those enhancements are not found in other Card-based Access Control Systems for vehicles in Truck-Loading Fuels Terminals.

5.2. Security Performance

Security of the access control system implies the protection of the terminal data against unauthorized access. This was enforced by the following design considerations:

- All the data was stored in a password-protected database server residing on the computer machine
- No entity from outside the computer machine was allowed to access the database directly

Moreover, the RFID ISO/IEC 14443 standard used, which was a unique enhancement, enforces authentication of RFID cards using secret keys before data reading or writing, hence increasing the level of security of the Access Control System [21]. Using cryptographic procedures and authentication with fixed keys offered greater security [21]. A successful authentication had to be performed first to allow any read or write memory operations [35].

5.3. Allocated Resources Performance

The resources utilized by an Online Access Control System mainly include:

1. the central processing unit (CPU) usage of the computer machine and
2. the amount of network traffic, i.e. the number of TCP/IP packets transmitted and received through the network.

For the sake of measuring both values, a number of experiments were carried out multiple times for a duration of around one hour and in each, the access control system (ACS) read the same number of access cards i.e. vehicles. The average results were calculated and compared for both: the traditional polling-based ACS and the developed event-driven IoT-based ACS.

5.3.1. Average CPU Usage (%)

The average CPU usage was measured using the task manager performance status of the operating system (OS). The measured results are shown in Figure 8. As shown in Figure 8, the developed IoT-based access control system is more performant than the traditional ones with regard to the CPU usage of the computer machine.

5.3.2. Average Number of TCP/IP Packets

The average number of the TCP/IP packets was measured through the *netstat* command which is used to display network protocols statistics for TCP/IP netwok connections [36].

It is clear from Figure 9 the very small number of TCP/IP packets transmitted and received in case of the developed IoT-based system compared to traditional access control systems, for the same number of access cards i.e. vehicles.

As seen from the comparison between the traditional and the developed access control systems listed in Table 2 that the allocated resources performance was enhanced drastically in the new design utilizing the IoT concepts. The practically measured results matched the theoretical design goals.

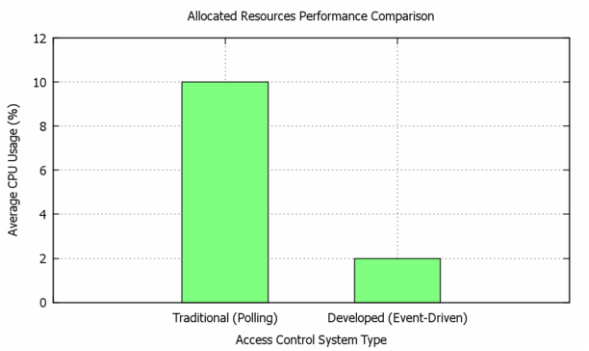

Figure 8. Comparison between Traditional ACS with Polling Server and the Developed Event-driven IoT-based ACS with regard to the average CPU usage.

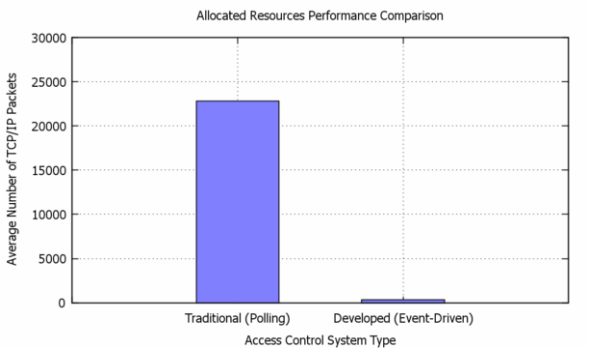

Figure 9. Comparison between Traditional ACS with Polling Server and the Developed Event-driven IoT-based ACS with regard to the average number of TCP/IP network packets.

Table 2. Comparison between Traditional ACS with Polling Server and the Developed Event-driven IoT-based ACS.

Point of Comparison	Traditional Polling ACS	Developed Event-Driven ACS
Average CPU Usage	~10%	~2%
Average Number of TCP/IP Network Packets	~22800	~370

6. Conclusion

In this paper, the notable IoT concepts were discussed and their applicable use-cases were presented. In addition, a developed IoT-based Online Access Control System for vehicles in Truck-Loading Fuels Terminals was presented. The advantages of the system design in comparison to traditional Card-based Access Control Systems for vehicles in Truck-Loading Fuels Terminals were stated with practical results. The Internet of Things not only connects the Cyber and Physical worlds, but also may enhance the performance of existing systems in various fields. It should be noted that the work done in this paper, and to the extent of the author's knowledge, applied a new approach in developing and studying the presented topic.

Acknowledgment

Warm thanks are due to the Communication and Electronics Department, the Faculty of Engineering of Alexandria University for the great input and support.

References

[1] ITU-T, TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU. Requirements and reference model of IoT-related crowdsourced systems. Geneva, Switzerland: International Telecommunication Union; 2019. Y.4205.
[2] Holler J, Tsiatsis V, Mulligan C, Avesand S, Karnouskos S, Boyle D. From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence. Elsevier Science ; 2014.
[3] Greengard, S . The Internet of Things. The MIT Press Essential Knowledge series. MIT Press ; 2015.
[4] Slama D, Puhlmann F, Morrish J, Bhatnagar RM. Enterprise IoT: Strategies and Best Practices for Connected Products and Services. O'Reilly Media ; 2015.
[5] International Telecommunication Union. ITU Internet Reports 2005: The Internet of Things. Geneva, Switzerland: International Telecommunication Union; 2005. 27441.
[6] Behmann F, Wu K. Collaborative Internet of Things (C-IoT): for Future Smart Connected Life and Business. Wiley - IEEE. Wiley ; 2015.
[7] Snigdha Sen. Internet of Things: an Introduction to Connecting the Unconnected. International Journal of Engineering Research and Technology (IJERT). 2016;4(29). ICIOT - 2016 Conference Proceedings.
[8] Sethi P, Sarangi SR. Internet of Things: Architectures, Protocols, and Applications. Journal of Electrical and Computer Engineering. 2017;2017. doi: 10.1155/2017/9324035.
[9] Kai Hwang, Geoffrey C Fox, Jack Dongarra. Distributed and Cloud Computing: From Parallel Processing to the Internet of Things. Elsevier Inc.; 2012.
[10] Rashmi. IoT (Internet of Things) Concept and Improved Layered Architecture. International Journal of Engineering Development and Research (IJEDR). 2018;6.
[11] Cruz-Piris L, Rivera D, Marsa-Maestre I, de la Hoz E, Velasco JR. Access Control Mechanism for IoT Environments Based on Modelling Communication Procedures as Resources. Sensors. 2018 March;18(917). doi: 10.3390/s18030917.
[12] William Stallings. Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud. Pearson Education ; 2015.
[13] COETZEE L, EKSTEEN J. The Internet of Things - Promise for the Future? An Introduction; 2011. IST-Africa 2011 Conference Proceedings. Paul Cunningham and Miriam Cunningham (Eds), IIMC International Information Management Corporation.
[14] Bonomi F, Milito R. Fog Computing and its Role in the Internet of Things. Proceedings of the first edition of the MCC workshop on Mobile Cloud Computing. 2012 August:13-6. doi: 10.1145/2342509.2342513.
[15] Brogi A, Forti S. QoS-aware Deployment of IoT Applications Through the Fog. IEEE Internet of Things Journal. 2017 May;4(5):1185-92. doi: 10.1109/JIOT.2017.2701408.
[16] McEwen A, Cassimally H. Designing the Internet of Things. Wiley ; 2013.
[17] William Stallings. Data and Computer Communications. 7th ed. Pearson Prentice Hall; 2004.
[18] Amazon Web Services, Inc (AWS). What is an Event-Driven Architecture?;. [Online; accessed 17-December-2022].
[19] Microsoft Azure. Event-driven architecture style;. [Online; accessed 17-December-2022].
[20] Zac Franken. Physical Access Control Systems; 2008. BlackHat DC.
[21] Klaus Finkenzeller. RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification. 2nd ed. John Wiley & Sons ; 2003.
[22] Modbus org. MODBUS over Serial Line Specification and Implementation Guide V1.02. Modbus-IDA; 2006.
[23] Piyu Dhaker. Introduction to SPI Interface; 2018. Analog Dialogue; Analog Devices Inc.
[24] Zurawski R. Industrial Communication Technology Handbook, Second Edition. Industrial Information Technology. Taylor & Francis ; 2014.
[25] Strauss C. Practical Electrical Network Automation and Communication Systems. Electronics & Electrical. Newnes ; 2003.

[26] Church P, Mueller H, Ryan C, Gogouvitis SV, Goscinski A, Tari Z. Migration of a SCADA system to IaaS clouds - a case study. Journal of Cloud Computing: Advances, Systems and Applications. 2017 June;6(11). Springer Open Access. doi: 10.1186/s13677-017-0080-5.

[27] Sun X, Yamada T, Takemura Y. Output Characteristics and Circuit Modeling of Wiegand Sensor. Sensors. 2019;19(13). Yokohama National University, Yokohama 240-8501, Japan. doi: 10.3390/s19132991.

[28] Brandon Chung. Wiegand Protocol Access: A Decade of Decryption; 2017.

[29] HID Corporation. Understanding Card Data Formats; 2006. Technology Basics White Paper.

[30] Benjamin Bucklin Brown. Over-the-Air (OTA) Updates in Embedded Microcontroller Applications: Design Trade-Offs and Lessons Learned; 2018. Analog Dialogue; Analog Devices Inc.

[31] Atmel Corporation. ATmega48A/PA/88A/PA/168A/PA/328/P: Atmel 8-bit Microcontroller with 4/8/16/32KBytes In-System Programmable Flash. 1600 Technology Drive, San Jose, CA 95110 USA; 2015.

[32] Armbrust M, Fox A, Griffith R, Joseph AD, Katz R, Konwinski A, et al. Above the Clouds: A Berkeley View of Cloud Computing. Electrical Engineering and Computer Sciences - University of California at Berkeley; 2009. UCB/EECS-2009-28.

[33] Reynders D, Mackay S, Wright E. Practical Industrial Data Communications: Best Practice Techniques. Practical professional books from Elsevier. Elsevier Science ; 2004.

[34] Charles E Spurgeon. Ethernet: The Definitive Guide. O'Reilly Media, Inc. ; 2000.

[35] NXP Semiconductors. MF1S50yyX/V1 MIFARE Classic EV1 1K - Mainstream contactless smart card IC for fast and easy solution development; 2014.

[36] Æleen Frisch. Windows 2000 Commands Pocket Reference. O'Reilly Media, Inc. ; 2001.