Light-Weight Secure IoT Key Generator and Management

Siew Leong KAN* and Paul LOH RuenChze

Nanyang Polytechnic, Singapore

Submission: October 14, 2018; **Published:** November 09, 2018

***Corresponding author:** Siew Leong KAN, Nanyang Polytechnic, Singapore

Abstract

Security is a critical element for IoT deployment that affects the adoption rate of IoT applications. This paper presents a Light-Weight Secure IoT Key Generator and Management Solution (LKGM) for industry automation and applications. Our solution uses minimum computing and memory resources that can be installed on half-credit-card-size embedded systems that enhances the security of end-to-end communications for IoT nodes. A frequently changed randomly generated passphrase is used to authenticate each IoT node that is embedded with an encrypted unique authentication key. Field test results were presented for an advanced manufacturing application that will only be activated when two authenticated IoT nodes are within the vicinity.

Keywords: Authentication; Authority; Secure key; IoT; Security; Industry automation

Introduction

Internet of Things (IoT) is a network of physical objects that have unique identifiers capable of producing and transmitting data across a network seamlessly. IoT system refers to a loosely coupled, decentralized system of devices augmented with sensing, processing, and network capabilities [1,2]. IoT is projected to be one of the fastest growing technology segments in the next 3 to 5 years [3]. IoT applications are being developed and deployed in an exponentially increasing manner in many smart city's initiatives around the world. Gartner Group has estimated that there will be 25 billion connected IoT devices by 2020, and that IoT services will constitute a total spending of \$263 billion. Unfortunately, this growth in connected devices brings increased security risks [4]. As indicated by Frost & Sullivan [5]; Miorandi et al., and Weber [6,7], security is the major hindrance for the wide scale adoption of IoT. In addition, the increasing use of multi-vendors IoT nodes which are often only have minimum security protection that resulted in more complex security scenarios and threats beyond the current Internet will arise. Constant sharing of information between "things" and users can occur without proper authentication and authorization. Currently, there are no trustworthy platforms that provide access control and personalized security policy based on users' needs and contexts across different types of "things". The "things" in any IoT network are often unattended; therefore, they are vulnerable to attacks. Moreover, most IoT communications are wireless that make eavesdropping easy [6,8]. The future widespread adoption of IoT will extend

the information security risks far more widely than the Internet has to date [9].

In an ad-hoc IoT network where IoT nodes are localized and self-organized, network infrastructure is not required. Security of the IoT nodes that operate in such ad-hoc peer-to-peer networks are increasingly becoming an important and critical challenge to solve as many applications in such IoT network becomes commercially viable. As ad-hoc IoT network has a frequently changing network topology, and the IoT nodes have limited processor power, memory size and battery power, a centralized security authentication server/node becomes impractical to be implemented.

Methods

In our applied research work, "KeyThings" was developed as part of the project title "Collaborative Cross-Layer Secure IoT Gateways" funded by the Singapore NRF-TRD. Our solution consisted of two main systems, namely the Security Key Generation System (SKG) and Security Key Management System (SKM). The objective of our project is to allow an IoT application (e.g. a web service, etc.) to be activated only when a pre-determined number of authenticated IoT nodes are within the vicinity. This enhances the security of the IoT application by authenticating the hardware (i.e. IoT nodes) instead of just authenticating based on the usual usernames and passwords. The authentication process is done in the system's background without the need for human intervention which is critical in

some operation environment (e.g. manufacturing, production, remote sites, etc.) where not all staff are given access to the sensors' readings due to security issues. The staff are categorized into "non-authorized", "operator" and "supervisor".

Below are the features of our solution

- a. "Non-authorized" personnel who are not issued with the authenticated IoT node will not have access to the sensors' readings.
- b. Only authorized "operator" who has an authenticated IoT node is able to view the sensors' readings only when the "operator" is in the vicinity.
- c. The authorized "supervisor" with an authenticated IoT node that is with higher access rights, can view the sensors' readings and the summary report. If the "supervisor" leaves the vicinity, the summary report will no longer be available.
- d. All authentications are done in the solution's background without the need for human intervention.

Solution Setup

Equipment (Figure 1)

Figure 1: The setup consists.

- A. The setup consists of the following equipment
 - a. Authentication Server
 - b. Client device 1
 - c. Client device 2
 - d. Application Server
 - e. Tablet

Authentication server (KeyThings-Server): The authentication server is the "brain" of the security key management. It has the following responsibilities:

- A. **Access point:** Serves as the access point to the entire system.

B. Generate random passphrase periodically

- i. If there is no authenticated device, the passphrase will remain the same.
- ii. If there is one or more authenticated device, a new random passphrase will be generated at the end of each time interval (after every 5 MQTT broadcasts).

C. MQTT Server: It will broadcast the generated passphrase via MQTT to all subscribed KeyThings-Clients.

- i. Once every 2 seconds.
- ii. MQTT topic: authentication/challenge

D. Web Server via REST API.

E. For KeyThings

- i. -clients to submit their encrypted passphrase.
- ii. For application server to query the number of authenticated devices.

F. **Authentication:** The server stores the encrypted credentials and MD5 of the KeyThings-Clients that were generated from the Security Key Generation System.

Client devices (KeyThings-Client): Each client device contains the unique security key that is used for authentication to gain access to 113 different web services. The key must be generated from the Security Key Generation System. The device has the following responsibilities:

A) **MQTT client.** Registers and listens to the broadcasted passphrase.

B) **Encryption.** Encrypts the passphrase that was received via MQTT.

a. If the received passphrase is the same as previous passphrase, the device will just ignore the passphrase and does nothing.

b. If the received passphrase is different from the previous passphrase, then the passphrase will be encrypted.

c. **HTTP Request / Response.** Send the encrypted passphrase to the authentication server (KeyThings-Server) once the encryption has been completed.

Application server: The application server hosts the production webpage (i.e. the machine readings and summary report). It is currently running on Raspberry Pi, but it can be hosted on any environment (i.e., Windows or Linux) that has network connectivity to the Access Point. The application server has the following responsibilities:

- a) **HTTP Request / Response:** Host the webpage that can be accessed via the tablet.

Tablet: The tablet is used to view the web page that contains the manufacturing data (machine readings and summary report) from the application server.

Result

Below is what you will see when different numbers of devices have been authenticated Figure 2.

Figure 2: Scenario 1: No device has been authenticated;
Scenario 2: Only one client device has been authenticated;
Scenario 3: Two client devices have been authenticated.

Discussion

The test was conducted successfully with results indicated that a light-weight security key generation and authentication method can be easily implemented in a distributed manner for a self-organizing network to enhance IoT nodes and service level security in an industry automation environment. The method and the solution can be applied to provide features such as multi-level security for different stake holders in an advanced manufacturing environment, multi-factor security keys, user definable security- based services and policy, etc. The solutions can easily be scaled and adapted to suite various industry needs and expectation in enhancing the security of IoT nodes, sensors, PLC controllers, robots, etc. to meet their business needs.

Conclusion

In this paper, a Light-Weight Secure IoT Key Generator and Management Solution (LKGM) for industry automation

and applications for enhancing the security of peer-to-peer communications among IoT nodes is presented. The LKGM is integrated to half-credit-card-size embedded systems. Our experimental results showed that the solution enhances secured peer-to-peer IoT communications amongst the IoT node. Field tests were conducted successfully for a manufacturing application that uses web services.

Funding

This research was funded by the Singapore National Research Foundation – Translational Research and Development (NRF-TRD) grant number MOE2015-TIF-2-T-DO4.

Acknowledgment

The authors would like to thank the Singapore National Research Foundation for providing the funding available for this research. This material is based on proof-of-concept and translational research projects supported by the Singapore National Research Foundation. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessary reflect the views of the Singapore National Research Foundation.

References

1. Li S, Li DX (2015) The Internet of Things: A Survey. Information Systems Frontiers 17(2): 243-259.
2. Al-Fuqaha A, Guizani M, Mohammadi M, Aledhari M, Ayyash M, et al. (2015) Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. IEEE Communications Surveys & Tutorials 17: 2347-2376.
3. Internet of Things (IoT) & Machine-To-Machine (M2M) Communication Market by Technologies & Platforms (RFID, Sensor Nodes, Gateways, Cloud Management, NFC, CEP, SCADA, ZigBee), M2M Connections, IoT Components - Worldwide Market Forecasts (2014 - 2019).
4. An Internet of Things Reference Architecture (2005) Symantec, California, United States, p. 22.
5. Sullivan F (2012) Internet of Things-Technology Penetration and Roadmapping Technical Insights.
6. Miorandi D, Sicari S, De Pellegrini F, Chlamtac I (2012) Internet of things: Vision, applications and research challenges. Ad Hoc Networks 10(7): 1497-1516.
7. Weber RH (2010) Internet of Things-New security and privacy challenges. Computer Law & Security Review 26: 23-30.
8. Atzori L, Iera A, Morabito G (2010) The internet of things: A survey. Computer Networks 54: 2787-2805.
9. Sicare S, Rizzardi A, Coen-Porisini A, Grieco LA (2015) Security, Privacy and Trust in Internet of Things: The Road Ahead. Computer Networks 76: 146-164.

This work is licensed under Creative Commons Attribution 4.0 License
DOI: [10.19080/RAEJ.2018.03.555625](https://doi.org/10.19080/RAEJ.2018.03.555625)

**Your next submission with Juniper Publishers
will reach you the below assets**

- Quality Editorial service
- Swift Peer Review
- Reprints availability
- E-prints Service
- Manuscript Podcast for convenient understanding
- Global attainment for your research
- Manuscript accessibility in different formats
(Pdf, E-pub, Full Text, Audio)
- Unceasing customer service

Track the below URL for one-step submission
<https://juniperpublishers.com/online-submission.php>