Bring Your Own Key for the Industrial Internet of Things

Thomas Ulz, Thomas Pieber, Christian Steger
Institute for Technical Informatics
Graz University of Technology
Graz, Austria
{thomas.ulz, thomas.pieber, steger}@tugraz.at

Sarah Haas, Holger Bock, Rainer Matischek
Development Center Graz
Infineon Technologies Austria AG
Graz, Austria
{sarah.haas, holger.bock, rainer.matischek}@infineon.com

***Abstract*—High tech strategies such as Industry 4.0 and Smart Manufacturing require industrial devices to be connected to the Internet. This movement towards interconnected industrial devices poses significant security risks as confidential data must be transferred and stored using untrustworthy channels and cloud servers. End-to-end private key cryptography is suitable to protect the confidentiality, integrity, and authenticity of data. However, private key cryptography has some drawbacks such as the so-called key distribution problem. A possible solution, factory installed keys, are untrustworthy as the two partners relying on end-to-end cryptography can not be sure that no other party is in possession of the used keys. To overcome these problems, the Bring Your Own Key (BYOK) principle based on Near Field Communication (NFC) and dedicated secured hardware is presented in this paper.**

***Index Terms*—Near Field Communication; Industrial Internet of Things; Industry 4.0; cryptography; keys; security controller.**

I. Introduction

The growth of the Industrial Internet of Things (IIoT) is driven by initiatives such as Industry 4.0 [1], Smart Manufacturing [2] or Cloud Manufacturing [3]. All of these initiatives promote the connection of production relevant devices to the Internet to quickly respond to changing customer demands, making them so-called smart factories [4]. Data in such smart factories does not only need to be transferred internally, but also to external partners to increase operational efficiency. Smart factories aim at increasing the operational efficiency through (i) minimizing unplanned downtime of production relevant equipment, (ii) improving the supply chain efficiency,

To decrease downtimes, the necessary maintenance, repair, and operations (MRO) schedules need to be optimized. Maintenance providers, for example, device vendords need to collect and analyze data such as equipment condition or operating hours [5] in order to predict optimal MRO schedules.

To increase supply chain efficiency, it is crucial to use Internet technologies and business-to-business supply chain applications [6]. In an IIoT context this includes the transmission of production data directly to suppliers such that the overhead of supply chain management can be minimized.

To be able to optimally monitor and control the internal production flow, a suitable smart factory architecture as well as protocols need to be chosen. A possible IoT protocol that is also suitable for industrial use cases is the Message Queue Telemetry Transport (MQTT) protocol that was designed for lightweight machine-to-machine communication [7]. MQTT is based on the publish/subscribe principle and through its architecture it is possible to transport data to internal as well as external partners such as maintenance providers [8].

In order to be able to transfer data to external communication partners such as maintenance providers and suppliers as well as to arbitrary internal devices, an MQTT broker that is connected to the Internet or even hosted by a third party can be used. In any case, the transport of confidential and production relevant information through the Internet requires the usage of appropriate cryptographic methods such as end-to-end encryption using TLS.

End-to-end encryption relying on asymmetric cryptography is infeasible for larger amounts of data; therefore, symmetric key cryptography needs to be used. Symmetric cryptography requires both the sender and receiver of the data to be in possession of the same shared key. Because no direct connection between sender and receiver can be established in protocols such as MQTT, key exchange algorithms such as Diffie-Hellman can not be used; a key distribution problem results from this scenario. A device vendor that is in possession of factory installed keys would need to distribute these keys to the equipment customer, who then would need to distribute some of these keys to its suppliers in a scenario such as illustrated in Fig. 1. Moreover, in such a scenario the device vendor would need to be trusted to securely and trustworthy handle all keys. For instance, if the device vendor would be selling devices to competing companies in the same business field, the device vendor would be in possession of keys that could be used for industrial espionage.

To mitigate these key related issues, we propose to apply the Bring Your Own Key (BYOK) principle. Using this principle, keys necessary for end-to-end encryption can be changed by device owners. For example, in the scenario illustrated in Fig. 1, the keys used to protect production relevant data can be changed such that the device vendor is no longer in possession of decryption keys for production data. To allow keys to be deployed and updated in a secured but intuitive manner, we present an NFC based approach that also uses dedicated security controllers (SC) to increase the security of our approach. To the best knowledge of the authors,

no such approach was presented previously. Therefore, the contributions of this paper are as follows:

- We present a scenario in that the BYOK principle is applied to solve the problems arising from key distribution and trust issues in factory deployed keys.
- We present a secured and NFC based interface for IIoT devices to deploy and change keys.
- The presented NFC extension for IIoT devices is suitable for new devices and legacy devices alike.

The remainder of this paper is structured as follows. All involved technologies as well as related work to the BYOK principle are discussed in Section II. In Section III, our approach to change keys in an IIoT context is presented. The security implications of the proposed BYOK approach for manufacturing devices are then analyzed by means of a threat analysis in Section IV. A prototypical implementation of our approach is shown in Section V. In Section VI this paper is concluded and possible future work is discussed.

II. Background and Related Work

A. Near Field Communication (NFC)

NFC is a wireless communication technique that is based on a subset of radio-frequency identification (RFID) standards. Because NFC is based on RFID standards, NFC devices are compatible with existing RFID cards and tags [9]. NFC technology is based on inductive coupling and operates at a radio frequency of 13.56 MHz up to a range of approximately 10 centimeters with bit rates up to 848 kbits per second [10]. Connecting NFC devices is fundamentaly different than other technologies such as WiFi, Bluetooth or ZigBee. Two devices automatically establish an NFC connection if they are brought near to each other. Thus, connections need to be (i) actively initiated by a human operator and (ii) the operator typically needs to be in close proximity to the devices. On the one hand, NFC offers security advantages compared to other wireless technologies [11] because of these properties. On the other hand, bringing one device near to another to transfer data is an easy and intuitive principle for humans [12]. In addition, NFC devices can be operated in passive mode which allows NFC devices such as tags or contactless cards to be operated without a battery or power supply [13].

NFC is seen as a promising IoT technology that will *link the real world with the digital world* [14]. Nowadays, NFC (or RFID) is already used for a wide range of applications, the most prominent being the mobile payment sector [15]–[17]. Other application domains of NFC include ticketing [18]–[20], healthcare [21]–[23], or pairing of wireless devices [24], [25].

B. Authenticated Encryption (AE)

AE combines *symmetric cryptography* with *Message Authentication Codes (MAC)* in a secured way such that data confidentiality, integrity, and authenticity can be provided [30]. Symmetric cryptography relies on a shared key for encryption and decryption of data [31]. In our presented approach, the *Advanced Encryption Standard (AES)* is used that is considered to be cryptographically secure using keylenghts of 256 bit [32].

TABLE I
Comparison with related work

Work	Method	Remark
[24]–[26]	Pairing of wireless devices	Approaches provide no or only weak security as information for device pairing is not considered confidential.
[27]	TLS secured key exchange between smart cards	Proposed method for EAP-TLS enabled smart cards. This approach is not suitable for IIoT devices.
[28], [29]	Android device as NFC gateway to Internet	Internet access necessary which might not be possible in all industrial settings. Also, man-in-the-middle attacks could be performed on gateway.
Our approach	NFC device for secured key transport	Security properties discussed in threat analysis.

As MAC algorithm, a *keyed-hash message authentication code (HMAC)* [33] based on SHA-256 is used.

C. Security Controller (SC)

In our presented approach SC are used to offer a protected processing environment as well as secured storage for the transferred keys. SC can be embedded into systems similar to traditional processing units [34]. The property that distinguishes SC from conventional processing units is *tamper resistance* [35]. SC that provide tamper resistance mitigate physical attacks by using appropriate countermeasures that are tested by the *Common Criteria (CC) for Information Technology Security Evaluation* [36].

D. Bring Your Own Key (BYOK)

The BYOK principle originated from the *Bring Your Own Device (BYOD)* idea that allowed employees to use their own mobile phones, tablets and laptops in company networks [37]. These devices need to be secured such that they can be trusted to access a company's confidential data [38].

Similar to BYOD, the BYOK principle allows own keys to be used for cryptographic operations [39], [40]. BYOK is mostly associated with cloud computing, where data is end-to-end encrypted using keys provided by the customer. If in addition to keys also cryptographic methods are provided by a customer, the BYOK principle is extended to *Bring Your Own Encryption (BYOE)* [41].

The establishment of an end-to-end secured channel using keys provided by a BYOK method could be interpreted as a device pairing process as well. The pairing of wireless devices is often assisted by NFC technology [24]–[26]. Urien et al. [27] present an approach to securely exchange tokens between smart cards used in prepayment contexts. Related to keys, Urien and Kiennert [28], [29] introduce an NFC based system to update access authorizations of RFID locks. In their approach they use Android mobile phones to establish a Internet connection via NFC that is used to download keys from a key server to the RFID lock. The Internet connection required in this approach however can be a drawback because

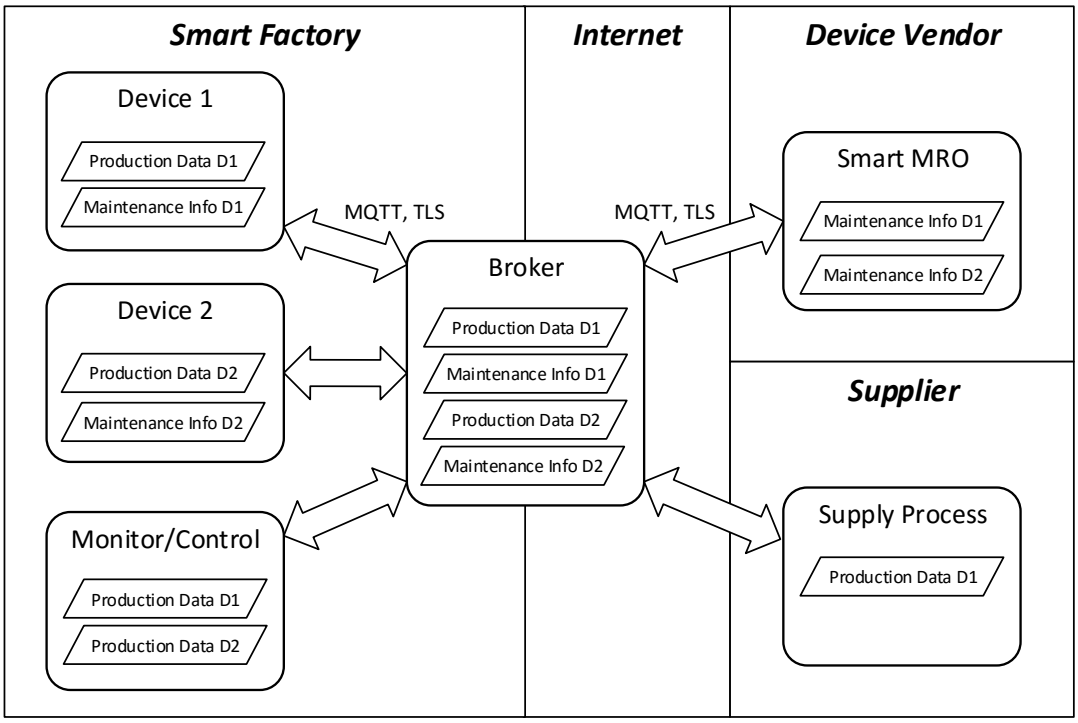

Fig. 1. Example of a smart factory with various publishers and subsribers of data, as envisioned in the Industry 4.0 initiative.

no configuration device tailored for IIoT use cases without network capabilities can be utilized. An overview of related work compared to our presented approach is given in Table I.

III. Bring Your Own Key

In general, the keys that are deployed using the BYOK principle need to be generated first. We propose two different scenarios to generate keys, depending on the trustworthiness of the used mobile device and the corresponding operator. Both scenarios are shown in Fig. 2.

1) If the mobile device and/or the personnel deploying keys are considered untrustworthy, keys are generated at a backend. The key material is then encrypted and transferred to the mobile device, from where the keys can be deployed at the manufacturing devices and the corresponding connection partners. The keys are protected from being extracted and used by an adversary due to the applied encryption.
2) If the mobile device *and* the personnel deploying keys are considered trustworthy, keys can be generated and encrypted directly at the mobile device. The key material then needs to be transferred to the manufacturing device and the corresponding connection partners.

NFC technology is used to transfer key material between devices in our approach. However, NFC does not provide cryptographic protection for transferred data. Therefore, we protect keys transferred in our NFC based BYOK approach by using AE to provide confidentiality, integrity and authenticity for these transferred keys. To encrypt and decrypt data using

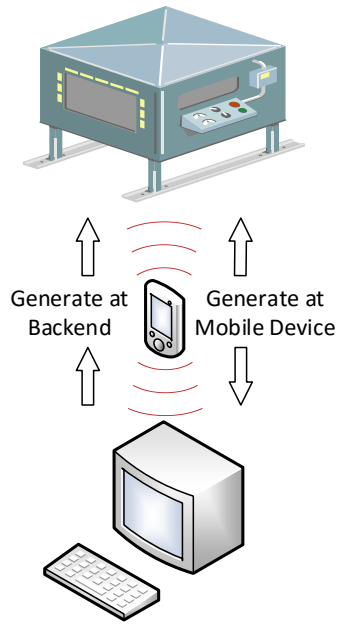

Fig. 2. Deploying new keys at manufacturing device and backend. The keys are either generated at the backend or at the mobile device itself.

AE, an initial key needs to be defined. If this is done by the equipment vendor, these key needs to be send to the equipment customer using a trusted channel. The equipment customer can change these initial keys immediately after delivery of

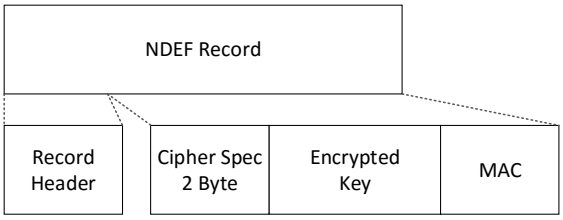

Fig. 3. NDEF Record containing a single transferred key protected using AE.

equipment from the vendor using the BYOK approach. Thus, the device customer is able to take control of their hardware. New keys protected by AE are transferred to the manufacturing equipment using NFC and the NFC Data Exchange Format (NDEF). NDEF packets can contain a number of NDEF Records that contain the actual data. In our approach, a single NDEF Record (see Fig. 3) contains the required header information as well as a cipher spec, the encrypted key and the MAC resulting from the AE encryption process. The cipher spec field contains information on which algorithms to use for decryption and MAC calculation. Encrypted key and MAC are sent sequentially, because the encrypt-then-MAC mode of operation was selected by us due to its security properties [42].

We propose the hardware extension for IIoT devices shown in Fig. 4, to provide the required NFC functionality as well as secured storage and execution environments for manufacturing devices. A host controller is used to connect manufacturing devices to the Internet by providing interfaces to the manufacturing device itself as well as to the Internet. In addition to that host controller, we propose to include a SC that provides an NFC interface as well as tamper resistance. The NFC interface is used to transfer keys from the mobile device to the manufacturing device. In addition to that, the SC can be powered through the NFC field, such that keys can be exchanged even if the manufacturing device is not connected to any power supply. The transferred keys are then decrypted and securely stored in the SC's memory that provides tamper resistance. Thus, it is infeasible for adversaries to extract keys transferred to and stored at the manufacturing device. The SC further provides tamper resistance for the cryptographic operations necessary during end-to-end encrypted data transfer via the Internet.

IV. Threat Analysis

A threat analysis [43] was conducted to highlight security features and to demonstrate the achieved security level of our presented BYOK approach. This threat analysis lists all involved **Entities (E)**, **Assets (A)** that need to be protected, and **Threats (T)** resulting from our BYOK approach as well as **Countermeasures (C)**, **Residual Risks (R)** and **Assumptions (As)** regarding the threats. For all involved entities, assumptions regarding their trustworthiness are made.

- (E1) *Device Vendor:* (As1) assumed honest but curious
- (E2) *SC Vendor:* (As2) assumed trustworthy
- (E3) *Device Owner:* (As3) assumed trustworthy

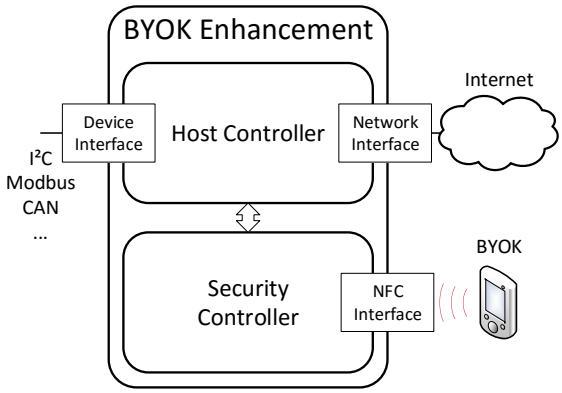

Fig. 4. BOYK enhancement providing interfaces to manufacturing device, the Internet and the mobile device used for key deployment.

- (E4) *External Communication Partners* (e.g. Supplier): (As4) assumed trustworthy
- (E5) *Hoster for Broker:* (As5) assumed untrustworthy
- (E6) *Person deploying Key:* (As6) assumed untrustworthy
- (E7) *Arbitrary Adversary:* (As7) assumed to be able to conduct online and physical attacks

After all entities and the corresponding assumptions are identified, the assets that need to be protected are determined.

- (A1) *Encrypted Data:* The data that is confidential and thus transferred secured by some key.
- (A2) *Keys:* All keys that are stored at any instance in the BYOK process. Loss of a key would result in a loss of confidentiality, integrity and authenticity of (A1).
- (A3) *Device Functionality:* A BYOK interface that is integrated into IIoT devices must not threaten the functionality of these devices in any way. If an adversary is able to harm the functionality of an IIoT device, physical entities and even human lives are threatened.

Considering all identified entities, assumptions, and assets, our presented BYOK approach can now be reviewed concerning potential threats. For each threat, we are going to list countermeasures and/or residual risks if a threat can not be mitigated. For each threat, the involved entities as well as the affected assets are listed as well.

- (T1) *Intentional or unintentional backdoors in device.*
  Entities/Assets: (E1), (E2); (A1), (A2), (A3)
  (C1) Threats investigated in CC EAL5+ certification process for the SC included in involved devices.
- (T2) *Weak or buggy cryptography.*
  Entities/Assets: (E1), (E2); (A1), (A2), (A3)
  (C2) Threats investigated in CC EAL5+ certification process for the SC included in involved devices.
- (T3) *Device vendor loses or distributes keys.*
  Entities/Assets: (E1); (A1), (A2), (A3)
  (C3) Initial keys are changed through BYOK approach. Device vendor does not own actually used keys.
- (T4) *Malicious mobile device or personnel.*
  Entities/Assets: (E6); (A1), (A2), (A3)

(C4) Key material is transported protected by AE, if personnel and/or device are assumed to be untrustworthy.

- (T5) *Wrong keys deployed.*
  Entities/Assets: (E6); (A3)
  (R1) A malicious user that deploys wrong keys or does not update keys and thus attacks communication between devices is similar to a DoS attack that can not be mitigated by our approach.
- (T6) *Device owner does not change initial keys, uses weak keys or loses keys in a security breach.*
  Entities/Assets: (E3); (A1), (A2), (A3)
  (R2) Malicious behaviour by the device owner can not be mitigated by our approach.
- (T7) *Remote attacks targeting IIoT devices.*
  Entities/Assets: (E7); (A1), (A2), (A3)
  (C5) Due to the short communication range of NFC, remote attacks are limited to attackers having physical access to a smart factory.
  (C6) An adversary that is able to communicate using the NFC interface is still not able to apply keys because the encryption key is kept private by the device owner.
  (C7) To mitigate the problem of eavesdropping that is still possible for any wireless technology, the transferred keys are protected using AE.
- (T8) *Physical attacks targeting IIoT devices.*
  Entities/Assets: (E7); (A1), (A2), (A3)
  (C8) Due to the SC providing tamper resistance, extracting key material is considered infeasible for adversaries.
- (T9) *DoS attack using BYOK interface.*
  Entities/Assets: (E7); (A3)
  (C9) Traditional DoS attacks using the BYOK interface are mitigated by the limited bit rate of NFC and the SC handling all involved cryptographic methods. Thus, the whole computational effort will be handled by the SC.

V. Prototype

A prototypical implementation of an end-to-end encrypted data transfer relying on keys provided through our presented BYOK enhancement was implemented to demonstrate the functionality, feasibility and usability of our approach. The setup consists of a mboile device and three Raspberry PI 3, representing a manufacturing device equipped with our BYOK enhancement, a broker and a subscriber respectively as shown in Fig. 5. Similar to the scenario shown in Fig. 1, the manufacturing device is connected to the smart factory's internal network (blue network cable) while the MQTT broker and the supplier are in an external network (yellow network cable). The used mobile device is a Nexus S smart phone with Android 4.1.2 Jelly Bean installed. The BYOK enhancement comprises the following components:

- The used host controller is an Infineon XMC4500 microcontroller from the Cortex M4 family that offers various connection interfaces such as USB, I2C and Ethernet.
- The SC is connected via I2C to the host controller. In our prototype we used an Infineon SLE78 that is CC EAL5+ (high) certified [36] as SC. This SC includes an NFC interface that is able to power the SC and connected devices such as sensors through the NFC field emitted by active NFC devices.

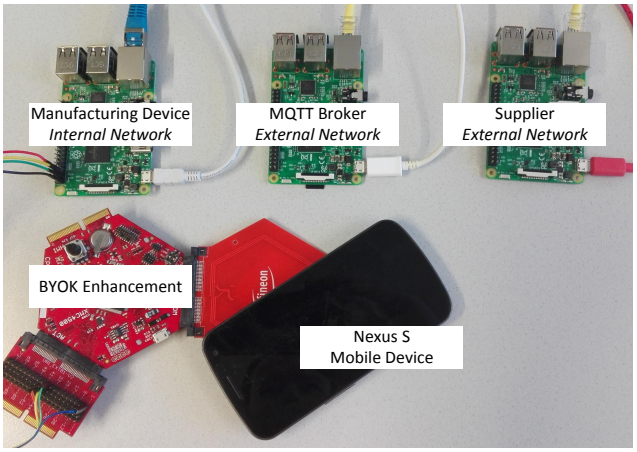

Fig. 5. Prototype setup using a Nexus S mobile device and three Raspberry PI 3, as well as our proposed BYOK enhancement.

VI. Conclusion and Future Work

In this paper we have shown how to apply the BYOK principle to mitigate key related problems arising in the IIoT. This principle, usually applied in cloud computing scenarios, assists in establishing end-to-end encrypted data transfers using IoT protocols such as MQTT. By enabling device owners to change factory deployed keys, this approach helps to increase trust in publishing manufacturing relevant confidential data to the Internet. Using NFC technology to transfer keys is intuitive and offers security advantages compared to other wireless technologies. The proposed BYOK hardware extension allows keys to be deployed using NFC in a secured manner, even if the manufacturing device is without a power supply. We have shown a prototype that highlights the functionality and feasibility of our approach. The presented approach is also shown to be secured against issues that would arise due to including an additional interface into manufacturing devices.

As future work we plan to extend our approach to not only support key material but arbitrary configuration data. As deploying malicious configuration data could lead to physical damage or even threaten human lives, security of transferred configuration data needs to be further improved compared to our current approach.

Acknowledgment

This project has received funding from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement No 692480. This Joint Undertaking receives support from the European Unions Horizon 2020 research and innovation programme and Germany, Netherlands, Spain, Austria, Belgium, Slovakia.

REFERENCES

[1] T. Bauernhansl, M. Ten Hompel, and B. Vogel-Heuser, *Industrie 4.0 in Produktion, Automatisierung und Logistik: Anwendung· Technologien· Migration*. Springer-Verlag, 2014.

[2] J. Davis, T. Edgar, J. Porter, J. Bernaden, and M. Sarli, "Smart manufacturing, manufacturing intelligence and demand-dynamic performance," *Computers & Chemical Engineering*, vol. 47, pp. 145–156, 2012.

[3] X. Xu, "From cloud computing to cloud manufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 28, no. 1, pp. 75–86, 2012.

[4] D. Lucke, C. Constantinescu, and E. Westkämper, "Smart Factory - A Step towards the Next Generation of Manufacturing," in *Manufacturing Systems and Technologies for the New Frontier*. Springer, 2008, pp. 115–118.

[5] A. H. Tsang, "Strategic dimensions of maintenance management," *Journal of Quality in Maintenance Engineering*, vol. 8, no. 1, pp. 7–39, 2002.

[6] R. A. Lancioni, M. F. Smith, and T. A. Oliva, "The Role of the Internet in Supply Chain Management," *Industrial Marketing Management*, vol. 29, no. 1, pp. 45–56, 2000.

[7] A. Banks and R. Gupta, "MQTT Version 3.1. 1," *OASIS standard*, 2014.

[8] C. Lesjak, D. Hein, M. Hofmann, M. Maritsch, A. Aldrian, P. Priller, T. Ebner, T. Ruprechter, and G. Pregartner, "Securing Smart Maintenance Services: Hardware-Security and TLS for MQTT," in *2015 IEEE 13th International Conference on Industrial Informatics (INDIN)*. IEEE, 2015, pp. 1243–1250.

[9] G. Van Damme, K. Wouters, and B. Preneel, "Practical Experiences with NFC Security on mobile Phones," *Proceedings of the RFIDSec*, vol. 9, p. 27, 2009.

[10] R. Want, "Near Field Communication," *IEEE Pervasive Computing*, vol. 3, no. 10, pp. 4–7, 2011.

[11] E. Haselsteiner and K. Breitfuß, "Security in Near Field Communication (NFC)," in *Workshop on RFID security*, 2006, pp. 12–14.

[12] D. López-de Ipiña, J. I. Vazquez, and I. Jamardo, "Touch Computing: Simplifying Human to Environment Interaction through NFC Technology," *1as Jornadas Científicas sobre RFID*, vol. 21, 2007.

[13] H. Mika, H. Mikko, and Y.-o. Arto, "Practical implementations of passive and semi-passive NFC enabled sensors," in *Near Field Communication, 2009. NFC'09. First International Workshop on*. IEEE, 2009, pp. 69–74.

[14] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[15] J. Ondrus and Y. Pigneur, "An Assessment of NFC for Future Mobile Payment Systems," in *Management of Mobile Business, 2007. ICMB 2007. International Conference on the*. IEEE, 2007.

[16] M. Pasquet, J. Reynaud, and C. Rosenberger, "Secure Payment with NFC Mobile Phone in the SmartTouch Project," in *Collaborative Technologies and Systems, 2008. CTS 2008. International Symposium on*. IEEE, 2008, pp. 121–126.

[17] G. Van Damme, K. M. Wouters, H. Karahan, and B. Preneel, "Offline NFC Payments with Electronic Vouchers," in *Proceedings of the 1st ACM workshop on Networking, systems, and applications for mobile handhelds*. ACM, 2009, pp. 25–30.

[18] S. L. Ghìron, S. Sposato, C. M. Medaglia, and A. Moroni, "NFC Ticketing: A Prototype and Usability Test of an NFC-Based Virtual Ticketing Application," in *Near Field Communication, 2009. NFC'09. First International Workshop on*. IEEE, 2009, pp. 45–50.

[19] A. Juntunen, S. Luukkainen, and V. K. Tuunainen, "Deploying NFC Technology for Mobile Ticketing Services Identification of Critical Business Model Issues," in *Mobile Business and 2010 Ninth Global Mobility Roundtable (ICMB-GMR), 2010 Ninth International Conference on*. IEEE, 2010, pp. 82–90.

[20] J. Neefs, F. Schrooyen, J. Doggen, and K. Renckens, "Paper Ticketing vs. Electronic Ticketing Based on Off-Line System 'Tapango'," in *Near Field Communication (NFC), 2010 Second International Workshop on*. IEEE, 2010, pp. 3–8.

[21] J. Bravo, D. López-De-Ipiña, C. Fuentes, R. Hervás, R. Peña, M. Vergara, and G. Casero, "Enabling NFC Technology for Supporting Chronic Diseases: A Proposal for Alzheimer Caregivers," in *European Conference on Ambient Intelligence*. Springer, 2008, pp. 109–125.

[22] R. Iglesias, J. Parra, C. Cruces, and N. G. de Segura, "Experiencing NFC-based Touch for Home Healthcare," in *Proceedings of the 2nd international conference on pervasive technologies related to assistive environments*. ACM, 2009, p. 27.

[23] A. Marcus, G. Davidzon, D. Law, N. Verma, R. Fletcher, A. Khan, and L. Sarmenta, "Using NFC-Enabled Mobile Phones for Public Health in Developing Countries," in *Near Field Communication, 2009. NFC'09. First International Workshop on*. IEEE, 2009, pp. 30–35.

[24] E. Uzun, K. Karvonen, and N. Asokan, "Usability Analysis of Secure Pairing Methods," in *International Conference on Financial Cryptography and Data Security*. Springer, 2007, pp. 307–324.

[25] R. Steffen, J. Preissinger, T. Schöllermann, A. Müller, and I. Schnabel, "Near Field Communication (NFC) in an Automotive Environment," in *International Workshop on Near Field Communication*, 2010, pp. 15–20.

[26] L. Chen, G. Pan, and S. Li, "Touch-driven Interaction Between Physical Space and Cyberspace with NFC," in *Internet of Things (iThings/CPSCom), 2011 International Conference on and 4th International Conference on Cyber, Physical and Social Computing*. IEEE, 2011, pp. 258–265.

[27] P. Urien, M. Pasquet, and C. Kiennert, "A Breakthrough for Prepaid Payment: End to End Token Exchange and Management Using Secure SSL Channels Created by EAP-TLS Smart Cards," in *Collaboration Technologies and Systems (CTS), 2011 International Conference on*. IEEE, 2011, pp. 476–483.

[28] P. Urien and C. Kiennert, "A New Key Delivering Platform Based on NFC Enabled Android Phone and Dual Interfaces EAP-TLS Contactless Smartcards," in *International Conference on Mobile Computing, Applications, and Services*. Springer, 2011, pp. 387–394.

[29] ——, "A New Keying System for RFID Lock Based on SSL Dual Interface NFC Chips and Android Mobiles," in *2012 IEEE Consumer Communications and Networking Conference (CCNC)*. IEEE, 2012, pp. 42–43.

[30] M. Bellare and C. Namprempre, "Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm," in *International Conference on the Theory and Application of Cryptology and Information Security*. Springer, 2000, pp. 531–545.

[31] M. Bellare, A. Desai, E. Jokipii, and P. Rogaway, "A Concrete Security Treatment of Symmetric Encryption," in *Foundations of Computer Science, 1997. Proceedings., 38th Annual Symposium on*. IEEE, 1997, pp. 394–403.

[32] J. Daemen and V. Rijmen, *The Design of Rijndael: AES - The Advanced Encryption Standard*. Springer Science & Business Media, 2013.

[33] H. Krawczyk, M. Bellare, and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication," RFC 2104 (Informational), Internet Engineering Task Force, Feb. 1997.

[34] P. Kocher, R. Lee, G. McGraw, A. Raghunathan, and S. Moderator-Ravi, "Security as a New Dimension in Embedded System Design," in *Proceedings of the 41st annual Design Automation Conference*. ACM, 2004, pp. 753–760.

[35] S. Ravi, A. Raghunathan, and S. Chakradhar, "Tamper Resistance Mechanisms for Secure Embedded Systems," in *VLSI Design, 2004. Proceedings. 17th International Conference on*. IEEE, 2004, pp. 605–611.

[36] D. Mellado, E. Fernández-Medina, and M. Piattini, "A common criteria based security requirements engineering process for the development of secure information systems," *Computer standards & interfaces*, vol. 29, no. 2, pp. 244–253, 2007.

[37] G. Thomson, "BYOD: enabling the chaos," *Network Security*, vol. 2012, no. 2, pp. 5–8, 2012.

[38] A. M. French, C. Guo, and J. Shim, "Current Status, Issues, and Future of Bring Your Own Device (BYOD)," *Communications of the Association for Information Systems*, vol. 35, no. 10, pp. 191–197, 2014.

[39] H. Zhang, "Bring your own encryption: balancing security with practicality," *Network Security*, vol. 2015, no. 1, pp. 18–20, 2015.

[40] S. Syed and M. Ussenaiah, "The Rise of Bring Your Own Encryption (BYOE) for Secure Data Storage in Cloud Databases ," in *Green Computing and Internet of Things (ICGCIoT), 2015 International Conference on*. IEEE, 2015, pp. 1463–1468.

[41] S. McGrath, "The Rise Of Bring Your Own Encryption - InformationWeek," http://www.informationweek.com/interop/the-rise-of-bring-your-own-encryption-/a/d-id/1320796, 9 2015, (Accessed on 12/28/2016).

[42] H. Krawczyk, "The Order of Encryption and Authentication for Protecting Communications (or: How Secure Is SSL?))," in *Annual International Cryptology Conference*. Springer, 2001, pp. 310–331.

[43] S. Myagmar, A. J. Lee, and W. Yurcik, "Threat Modeling as a Basis for Security Requirements," in *Symposium on requirements engineering for information security (SREIS)*, vol. 2005. Citeseer, 2005, pp. 1–8.