Secured Offline Authentication on Industrial Mobile Robots using Biometric Data

Sarah Haas[1], Thomas Ulz[2], and Christian Steger[2]

[1] Development Center Graz, Infineon Technologies Austria AG, Graz/Austria
sarah.haas@infineon.com
[2] Institute for Technical Informatics, Graz University of Technology, Graz/Austria
{thomas.ulz, steger}@tugraz.at

Abstract. The increased usage of mobile robots in the industrial context entails higher safety risks for employees on the production floor. To enable safety, the usage of security concepts on industrial mobile robots is essential. One step towards security is authentication that is necessary to prevent unauthorized people from manipulating an industrial mobile robot's software or configuration. Traditional authentication schemes that utilize username and password are not feasible for industrial mobile robots as either (a) a remote connection would be necessary to check the credentials or (b) the credentials need to be checked locally on the robot. Remote connections are problematic due to connectivity problems similar to them at RoboCup Logistics League competitions. If the credentials need to be checked on the robot, the usernames and passwords of all authorized people need to be stored and maintained there. As both possibilities are not feasible for industrial mobile robots, we propose an offline authentication approach that uses biometric data to authenticate a user on a mobile robot. The approach uses expiring passwords and a smart card to authenticate authorized people on the mobile robot. The smart card is equipped with a fingerprint reader to check that only authorized people are allowed to authenticate at a mobile robot. To show that the approach is able to provide secured authentication, a threat analysis is performed.

1 Introduction

In the last decades, production facilities introduced automation to produce more products in less time and to increase the production efficiency and with that decrease costs. However, customer demands changed from traditional products to highly customized products. These customized products require the production of small batch sizes which cannot be achieved in traditional mass production facilities. This trend of customized products started with the beginning of the *fourth industrial revolution* [19]. In this context, movements such as *Industry 4.0* [2] try to develop concepts for future *smart factories* [22] to reinvent highly flexible production in industrial countries. One of the developments from the Industry 4.0 movement is the *cyber physical system* (CPS) [17]. A CPS is a

physical process controlled and monitored by some software and can even be connected to the Internet.

The increased automation in factories entailed an increased usage of robots for industrial applications which are a form of CPSs as they use sensor and actuators to interact with the physical world and are controlled by a computer system. Industrial robots for automation are widely used in manufacturing nowadays, but the importance of Industrial Mobile Robots (IMR) also increased in the last years due to their flexibility [18]. This flexibility can be seen in the RoboCup Logistics League Sponsored by Festo [16] (RCLL) where a production floor is simulated where IMRs need to fulfill tasks in order to manufacture products. The tasks in the RCLL are limited to transportation of material between machines that manipulate the material. These transportation tasks are a very realistic scenario also in real production facilities. In real production facilities, humans will also move on the production floor together with the IMRs similar to the referees in the RCLL. Due to the fact that IMRs and humans share the same environment, safety risks arise that are not present when material is moved using static conveyor systems. As IMRs are controlled by computer systems, a precondition for safety is the security of the computer system [12, 5].

One major step towards security is authentication where humans or machines need to prove their identity in order to get authorized to access a system. In the context of IMRs, authentication is a crucial topic as it is possible that external staff such as custodial or maintenance staff are present on the production floor. As a precaution, the external staff is assumed to be not trustworthy and might manipulate an IMR's interface or change its configuration. These manipulations might lead to safety threats for employees on the production floor. Therefore, authentication on the IMRs is necessary to prevent unauthorized people from performing malicious actions on an IMR. However, traditional username and password based authentication schemes are not feasible in this scenario as it is either necessary to

(a) remotely connect to a server to check the user credentials or

(b) to store the credentials locally on the robot to check them.

The remote connection might suffer from connectivity problems due to the wireless connection as it can be seen in the RCLL. Storing the credentials locally on the robot poses a high maintenance effort as the credentials need to be updated each time a new user is authorized to access the robot and each time a user's authorization is taken away. Both of these problems show that a traditional credential-based authentication is nearly infeasible for this scenario. Therefore, we propose a mechanism using one-time passwords, a smart card, and biometrics in the form of a fingerprint to authenticate users on IMRs.The one-time passwords in combination with a time value are used to generate passwords that expire after one usage to avoid security problems regarding revealed passwords. The smart card is used to generate and store the one-time passwords securely. The fingerprint is used to authenticate the user on the smart card to overcome issues with stolen or lost smart cards. The mentioned time needs to be

synchronized to enable the authentication. The synchronization is done by using the machines on the production floor as a gateway into the network to avoid the necessity of a direct wireless connection to a gateway.

To sum up, the contributions of this paper are:

- An authentication approach that allows users to authenticate on an IMR without the need for traditional credentials
- A method to synchronize the time using equipment on the production floor

The remainder of this paper is structured as follows. Section 2 discusses the related work and background regarding topics such as authentication and one-time-passwords. In Section 3, the authentication approach is shown in detail. Furthermore, the synchronization of the time, as well as the recovery behavior in case of a not synchronized time, are shown. In Section 4, a security analysis was made to show that the proposed authentication approach prevents unauthorized people from accessing the system. Finally, the paper is concluded in Section 5.

2 Background and Related Work

2.1 Authentication on Mobile Robots

To the best knowledge of the authors, the only existing user authentication approach using biometric data on robots was proposed by Kim et al. [8]. Kim et al. use a vision-based method to authenticate users by recognizing users faces, clothes colors and body height. The recognized biometrics are combined to identify users, but the method is also able to identify users when only the body height is available. However, each user's data needs to be stored on each robot to be able to identify them.

A more lightweight approach addresses authentication of mobile agents on other mobile agents [9]. The main differences to the previous approach are the use of shared secret keys instead of biometrics for authentication and that agents authenticate on other agents. The authors state that public key cryptography cannot be used in such authentication scenarios due to the fact that they are slow and due to the occurring overhead. The shared key in combination with some other properties such as the agent ID is used to create a ticket that can be verified with the same shared key. A ticket created once does not become invalid which might make it more vulnerable to attacks.

A very interesting approach was proposed by Wael Adi where some kind of electronic DNA (eDNA) is generated for each robot for authentication [1]. The author uses physically unclonable functions to create a unique eDNA for each robot. The eDNA is generated initially by key derivation and hashing of some initial secret keying material. Then an eDNA chain is generated by creating so-called identity modules that are used to identify a specific part of the robot. Using those modules, anyone can validate the identity of the robot by challenging the module. Later, the robots eDNA changes with its interactions. To authenticate, the robot needs to prove that it participated in a particular event that is stored

in its eDNA chain. This approach is very interesting but also very complex and requires knowledge of the whole domain a robot moves in to be able to verify the robot's identity.

2.2 Biometric Authentication

Clancy et al. [3] proposed a smart card based authentication algorithm that uses fingerprints to identify users. First, a fingerprint scanner on a terminal is used to capture a user's fingerprint. Then, a numeric template is generated from the fingerprint and sent to the smart card as well as a digit that should be signed. Afterward, the smart card verifies the template, generates a signature for the sent digit and sends it back to the terminal where the signature is verified, and the access is granted. The main contribution of this paper, however, was an efficient and more accurate algorithm to detect the fingerprints and decrease the number of false positives. Many approaches try to improve the quality of fingerprint checking such as [7], [20] or [21] to overcome issues with performance, false positives or attacks such as template attacks.

Li et al. [11] proposed an approach for biometric-based remote authentication of users using smart cards where both smart card and server need to prove their identity. In their approach, the user has to register on a server with his identity (i.e. username), password and biometric data (i.e. fingerprint). Afterward, the user receives a smart card that can be used for authentication. Both smart card and server send messages to each other containing random numbers. Both try to verify the message and authenticate each other if the message is correct.

2.3 Secure Element

A secure element is a hardware device that is used to store confidential or cryptographic data securely, and provides tamper-resistance. Secure elements might also be able to perform operations such as computing cryptographic hashes or encrypting data. These secure elements are used to protect any kind of confidential data against physical attacks. Examples are chip cards used for banking applications or trusted platform modules [10] in laptops to protect passwords or the firmware.

2.4 One-Time Passwords

The one-time password (OTP) concept was invented by Haller [6] in 1994 to overcome the issues with usernames and passwords that are transmitted in plain text and are therefore prone to eavesdropping attacks. The idea was to compute a hash of the credentials to prevent adversaries from revealing the plain text passwords. OTPs were later also considered as a solution to prevent replay attacks where an attacker captures the hashed or encrypted login information and repeats to send it to the targeted system later in time to gain access to the targeted system without knowing the actual login information. Replay attacks

were prevented by adding a moving factor to the hash such as a counter or time value.

M'Raihi et al. [13] proposed an HMAC-based one-time password algorithm (HOTP) that adds additional security to OTPs by introducing a counter to generate different OTPs even if the secret key or password is always the same. HMAC is a message authentication code that relies on cryptographic hash functions. To compute an OTP with HOTP, the secret key and the current counter are passed to the hash function of the HMAC. Afterward, the counter is increased by a specific value. The generated OTP is truncated and presented to the user on a token. The truncated OTP must then be typed into the system the user wants to authenticate on.

In contrast to HOTP that is an event-based algorithm where the counter changes with each newly generated OTP, TOTP is based on a time value instead of events [14]. The principle of TOTP is similar to HOTP which means that an HMAC using a secret key and a time value are used as parameters for the hash function. As already said, the moving factor for TOTP is a time value that increases continuously and does not depend on events. However, TOTP also requires a token that displays the truncated OTP to type it into the system a user wants to authenticate on.

3 Authentication Approach

The authentication approach proposed in this paper checks that only authorized people gain access to the IMRs. As traditional username and password based authentication approaches are not feasible in this scenario, the approach uses a smart card with an integrated fingerprint reader and a secure element on the IMR to authenticate a user. The proposed approach is based on TOTP [14] but does not use tokens and does not require the user to input a password into the IMR. To authenticate, the user only needs to place the smart card near the IMR's NFC module as it can be seen in Figure 1.

Furthermore, several preconditions need to be met to perform an authentication:

- **Backend:** System holding the serial numbers, secret keys and time values of each IMR. The backend is able to generate and store a derived key for each IMR. This is done by providing the IMR's serial number and secret key as parameters for a strong cryptographic hash function such as SHA-256 [4]. The reason for derived keys is that one single cryptographically strong master key can be used to generate a bunch of cryptographically strong derived keys. It also increases the security of the whole approach by using individual keys on every robot.
- **User:** Staff that has access to the backend to store the time values and derived keys on the smart card.
- **IMR:** Robot equipped with an NFC module with an LED for notifications and a secure element holding the derived key of the robot and increasing the time value.

– **Smart Card:** A secure element that stores the derived key and time value capable of computing an OTP. It is equipped with a fingerprint reader and an NFC antenna. The time value stored on the smart card is one that lies within the time range Δt when the user tries to authenticate on the IMR. Δt can be a range of several hours.

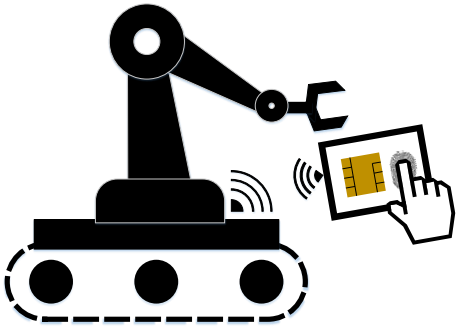

Fig. 1: Authentication of a user on a robot using NFC and a fingerprint reader on a smart card.

3.1 Biometric Authentication Approach

The authentication approach consists of two steps. In step one the user needs to authenticate himself on the smart card to unlock the computation of the OTP. Step two is to send the computed OTP to the IMR and verify it. In the following passage, the authentication will be described in detail with references to the numbers in Figure 2.

(1) Independent from any authentication or other operation, the IMR always computes batches of OTPs, meaning that the IMR computes a number of OTPs for a specific time range Δt that can be changed by authorized staff. The IMR might, for example, compute 36 OTPs for a Δt of 3 hours where one OTP is computed for every 5 minutes. The computation of batches of OTPs is done to make the verification of an OTP from a smart card possible as the smart card is not able to increase the time due to the fact that it can only compute OTPs when in the range of an NFC field. To compute the OTPs, the time value and the derived key are provided as parameters to a strong cryptographic hash function such as SHA-256. The output of the hash function represents the OTP. (2) The authentication is initiated by the user who places his finger on the fingerprint sensor and brings the smart card in the NFC field provided by the IMR. (3) The smart card verifies the fingerprint by comparing the input fingerprint with the one stored on the smart card's memory. (4) The smart card counts the authentication attempts that failed when users provided their fingerprint. If the number of failed authentication attempts is smaller than 10 and the provided fingerprint matches the one stored on the smart card, the OTP is computed. The OTP is computed by providing the IMR's derived key and the time value stored

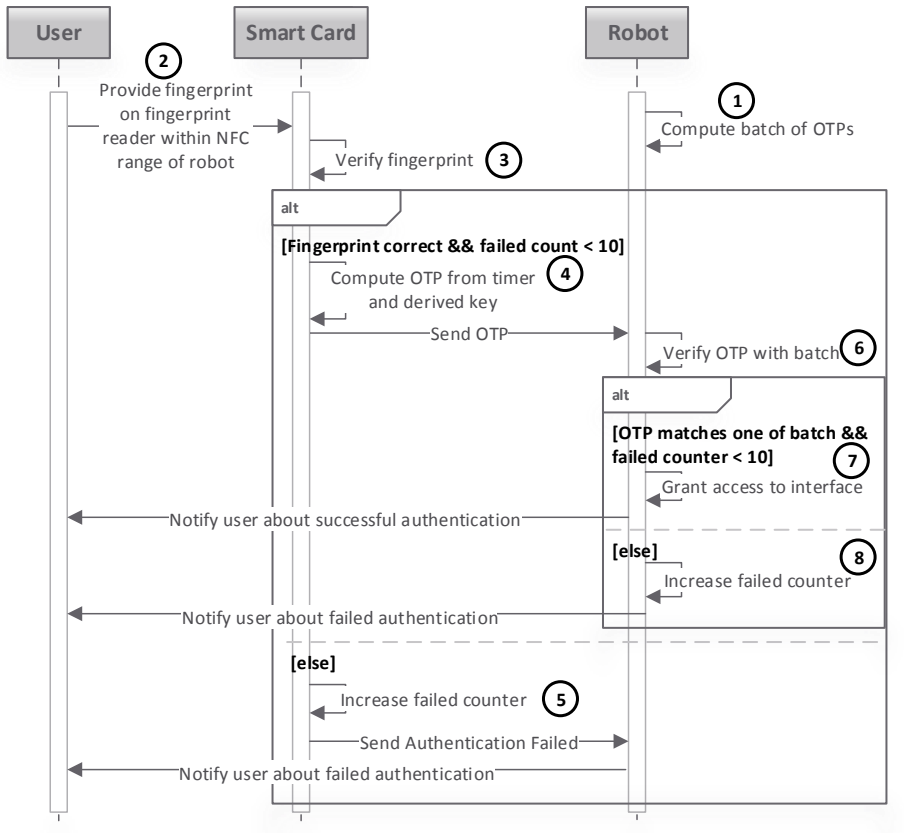

Fig. 2: Sequence diagram of the authentication approach.

on the smart card as parameters for the hash function. The hash is computed by the smart card and then sent to the IMR for verification. (5) In the case that the number of failed attempts exceeds 10 or the provided fingerprint does not match the one stored on the smart card, the number of failed authentications is increased by 1 and the authentication is not performed. The notification of a failed authentication is sent to the IMR to inform the user about the failed authentication attempt by the IMR using an LED that turns red. (6) The IMR verifies the OTP by comparing it to the batch-generated OTPs of the last time period Δt. (7) The IMR also counts the failed authentication attempts where the provided OTP was not valid. If the OTP from the smart card matches one of the batch-generated OTPs and the number of failed authentication attempts is below 10, the access to the IMR is granted to the user, and the LED on the IMR's NFC module turns green. (8) If the provided OTP by the smart card does not match one of the OTPs from the batch or if the number of failed authentication attempts exceeds 10, the number of failed authentications is increased, and the

user is notified about the failed authentication by changing the NFC module's LED to red.

3.2 Synchronization

The presented authentication approach uses a time value that is stored and increased by the IMR and also stored and increased on the backend. Even if the presented approach is an offline solution that does not need any connection to the backend for authenticating, the time values need to be synchronized as it is possible that they diverge with time due to the fact that the IMR and the backend do not rely on the same hardware. Therefore, the time values of each IMR need to be synchronized with the backend. To avoid the necessity of some wireless communication technique on the IMR, the synchronization is done using the equipment on the production floor that already has a wired network connection and the already equipped NFC module on the IMR. The machines on the production floor are stationary and therefore have a wired connection into the network. The synchronization is always done when an IMR communicates with a machine. The IMR encrypts its current time with his derived key and uses the machine to transfer it to the backend. The backend is able to decrypt the IMR's message as it is in possession of the IMR's derived key and stores the new time value.

3.3 Recovery Behavior

The synchronization behavior is necessary to ensure that the IMR's time and the backend's time are almost equal. However, it can happen that an IMR has a malfunction and is not able to synchronize. If the time values diverge in this case, it would not be possible to authenticate on the IMR anymore. Therefore, a recovery behavior is necessary to enable a user to authenticate on the IMR even if the times on the IMR and the backend diverge. The owner of the IMRs is provided with a recovery key that can be used to enable authentication even if the time values diverge. To recover the time value, the recovery key that is stored on a smart card is provided to the IMR. If the recovery key is correct, the LED on the NFC module turns green. Then the user must put the smart card holding the derived key of the IMR in the NFC range of the IMR. The IMR sends its current time value once to the provided smart card. With the value, the smart card can then perform the normal authentication steps. This recovery behavior is feasible as the user does not need to reinstall the time on the smart card over the backend. Furthermore, the recovery behavior is secured due to the fact that another smart card is necessary to reactivate the authentication. A condition, however, is that the user trying to gain access to the IMR is not the same user that holds the recovery key.

Another recovery needs to be done when the number of authentication attempts on the IMR is reached. This can also be done using the recovery key. The number of failed authentications is simply reset when the recovery key is

provided to the IMR. Afterward, the normal authentication process can be performed.

It is not necessary to define a recovery behavior for the smart card as disabling the smart card after too many attempts is a security mechanism. Furthermore, the smart card can be reinstalled with a correct derived key and a correct time value.

4 Evaluation

4.1 Comparison to other Approaches

Table 1 shows a feature comparison between the proposed approach and three state-of-the-art approaches described in Section 2. The table's cells are contain *Y*s and *N*s. Y indicates the presence of the feature, N indicates the feature's absence. Furthermore, the cells are marked red or green to show whether this feature's presence or absence in the approach indicates a weakness. Red means that the feature is a weakness, green means that the feature is a strength. The chosen features indicate the security level of the approach as well as the approach's feasibility in real life when authenticating offline. Expiring passwords make it significantly harder for adversaries to hack the robot. Two-factor authentication adds another level of security as a correct fingerprint and smart card are necessary. A recovery behavior increases the approach's feasibility. Individual user data on the robot is accompanied by a high maintenance effort when adding or removing authorized people. Offline authentication overcomes issues with wireless communication regarding connectivity.

The table shows that there is no approach yet that satisfies as many features as the one proposed in this paper.

	Kim et al. [9]	Kim et al. [8]	Li et al. [11]	This Paper
Expiring Password/Ticket	N	N	N	Y
Two-factor authentication	N	N	Y	Y
Recovery behavior	N	N	N	Y
Individual user data on robot	Y	Y	N	N
Offline authentication	Y	Y	N	Y

Table 1: Comparison of different state-of-the-art approaches with the proposed authentication approach.

4.2 Threat Analysis

To show the security level of the proposed approach, a threat analysis [15] was performed. A threat analysis is used to list **Entities (E)**, **Assets (A)** that need protection, possible **Threats (T)**, necessary **Assumptions (As)** as well as **Countermeasures (C)** and **Residual Risks (R)** that occur in the approach.

In the beginning, it is necessary to identify the entities and make assumptions regarding their trustworthiness. The entity **(E1)** User is assumed to be not trustworthy and might be a possible adversary. The entity **(E2)** Smart Card

is assumed to be trustworthy. The entity **(E3)** IMR is assumed to be curious. Entity (E4) secure element is assumed to be trustworthy. Entity **(E5)** malicious adversary is assumed to be not trustworthy and might be able to perform attacks on equipment.

Using the entities, the assets that need to be protected need to be discussed. **(A1)** Keys on the IMR and the smart card need to be protected. Loss of the keys might enable an adversary to reveal the time value and access an IMR. **(A2)** Time value should be protected as the loss might make attacks easier. **(A3)** Interfaces on the IMR need to be protected from unauthorized usage not to endanger employees.

Before discussing the threats, several assumptions regarding the authentication approach need to be stated. First, (As1) the IMR uses a secure element to store the derived key, and check and generate the OTPs. Second, (As2) the derived key was already stored on the IMR's secure element. Third, (As3) the time value is synchronized between IMR and backend. Fourth, (As4) it is assumed that the backend is sufficiently secured and therefore, the threat analysis will not address it.

For each threat, the assets and entities, as well as the possible countermeasures and residual risks, are listed. The residual risks are threats that cannot be mitigated by the proposed approach.

(T1) Intentional and unintentional backdoors in secure element and **(T2)** wrongly implemented/weak cryptography on the secure element both concern the entities (E2), (E3) and (E4) and threaten the assets (A1) and (A2). Both threats can be secured by (C1) as secure elements are certified for a specific security level. The certificate proves the correctness of the cryptographic implementation and that no undocumented backdoors exist.
(T2) Loss/theft of Smart Card concerns the entities (E1), (E2) and (E5) and threatens the assets (A1), (A2) and (A3). The smart card is protected (C2) due to the necessity to authenticate on the card using a fingerprint where an adversary cannot simply use the smart card for authentication on the IMR. The time value is (C3) as the data on the smart card expires, and the authentication cannot be performed afterward. (C4) Protects the derived key and time value as the smart card is tamper resistant. The limited authentication attempts are (C5) as it shuts down brute-force attacks.
(T4) Manipulation of the time value on smart card or IMR concerns entities (E1), (E2), (E3), (E4) and (E5), and threatens the asset (A2) and (A3). (C6) Protects the authentication as a wrong generates invalid OTPs. The time value is stored on a secure element that aims to protect the time value from any manipulations (C7).
(T6) Replay attacks with same OTP concern entities (E1), (E2), (E3), (E4) and (E5), and threatens the asset (A3). This threat is protected by (C8) as the OTP becomes invalid after the first usage due to the counter for several authentications with the same time value.
(T7) Physical attacks on IMR and smart card concerns entities (E2), (E3), (E4)

and (E5) and threatens the assets (A1), (A2) and (A3). The secure elements on IMR and smart card provide tamper resistance which means that adversaries are hindered from revealing the key (C9).
(T8) DoS attack on an IMR concerns entities (E1), (E3) and (E5) and threatens the asset (A3). The limited number of authentication attempts mitigates DoS attacks (C10).
(T9) Remote attacks on the authentication process concerns entities (E1), (E2), (E3) and (E5) and threatens the asset (A3). The short communication range of NFC limits the possibilities for remote attacks significantly (C11).

The given threat analysis is not exhaustive and addresses the most important threats identified by the authors. However, the analysis shows that no residual risks remain for the proposed approach.

5 Conclusion

In this work, an offline authentication approach for industrial mobile robots using a smart card and biometric data was proposed. The approach is used to overcome the connectivity problems with remote authentication schemes and the maintenance issues with traditional username and password based schemes. The approach uses a derived key and a time value to generate one-time passwords that expire and do not rely on a specific user that reduces the maintenance effort significantly compared to user dependent passwords. The evaluation emphasizes the security features of the proposed approach and shows the feasibility compared to other approaches.

Acknowledgment

The work/A part of the work has been performed in the project Power Semiconductor and Electronics Manufacturing 4.0 (SemI40), under grant agreement No 692466. The project is co-funded by grants from Austria, Germany, Italy, France, Portugal and - Electronic Component Systems for European Leadership Joint Undertaking (ECSEL JU).

References

1. Adi, W.: Mechatronic Security and Robot Authentication. In: 2009 Symposium on Bio-inspired Learning and Intelligent Systems for Security. pp. 77–82. IEEE (2009)
2. Bauernhansl, T., Ten Hompel, M., Vogel-Heuser, B.: Industrie 4.0 in Produktion, Automatisierung und Logistik: Anwendung· Technologien· Migration. Springer-Verlag (2014)
3. Clancy, T.C., Kiyavash, N., Lin, D.J.: Secure Smartcardbased Fingerprint Authentication. In: Proceedings of the 2003 ACM SIGMM Workshop on Biometrics Methods and Applications. pp. 45–52. WBMA '03, ACM (2003)
4. Gilbert, H., Handschuh, H.: Security Analysis of SHA-256 and Sisters. In: International Workshop on Selected Areas in Cryptography. pp. 175–193. Springer (2003)

5. Grieco, L.A., Rizzo, A., Colucci, S., Sicari, S., Piro, G., Di Paola, D., Boggia, G.: IoT-Aided Robotics Applications: Technological Implications, Target Domains and Open Issues. Computer Communications 54, 32–47 (2014)
6. Haller, N.: The S/KEY One-Time Password System. In: In Proceedings of the Internet Society Symposium on Network and Distributed Systems (1994)
7. Jin, A.T.B., Ling, D.N.C., Goh, A.: Biohashing: Two Factor Authentication Featuring Fingerprint Data and Tokenised Random Number. Pattern Recognition 37(11), 2245–2255 (2004)
8. Kim, D., Lee, J., Yoon, H.S., Cha, E.Y.: A Non-Cooperative User Authentication System in Robot Environments. IEEE Transactions on Consumer Electronics 53(2), 804–811 (2007)
9. Kim, J.G., Kim, G.S., Eom, Y.I.: Lightweight Mobile Agent Authentication Scheme for Home Network Environments. In: International Conference on Computational and Information Science. pp. 853–859. Springer (2004)
10. Kinney, S.L.: Trusted Platform Module Basics: Using TPM in Embedded Systems. Newnes (2006)
11. Li, C.T., Hwang, M.S.: An Efficient Biometrics-based Remote User Authentication Scheme using Smart Cards. Journal of Network and Computer Applications 33(1), 1 – 5 (2010)
12. Line, M.B., Nordland, O., Røstad, L., Tøndel, I.A.: Safety vs Security? In: Proc. 8th International Conference on Probabilistic Safety Assessment and Management (PSAM 2006), New Orleans, USA (2006)
13. M'Raihi, D., Bellare, M., Hoornaert, F., Naccache, D., Ranen, O.: HOTP: An HMAC-based One-Time Password Algorithm. Tech. rep. (2005), RFC4226
14. M'Raihi, D., Machani, S., Pei, M., Rydell, J.: TOTP: Time-based One-Time Password Algorithm. Tech. rep. (2011), RFC6238
15. Myagmar, S., Lee, A.J., Yurcik, W.: Threat Modeling as a Basis for Security Requirements. In: Symposium on requirements engineering for information security (SREIS). vol. 2005, pp. 1–8. Citeseer (2005)
16. Niemueller, T., Ewert, D., Reuter, S., Ferrein, A., Jeschke, S., Lakemeyer, G.: RoboCup Logistics League Sponsored by Festo: a Competitive Factory Automation Testbed. In: Automation, Communication and Cybernetics in Science and Engineering 2015/2016, pp. 605–618. Springer International Publishing (2016)
17. Rajkumar, R.R., Lee, I., Sha, L., Stankovic, J.: Cyber-Physical Systems: The Next Computing Revolution. In: Proceedings of the 47th Design Automation Conference. pp. 731–736. ACM (2010)
18. Schneider, S., Hegger, F., Hochgeschwender, N., Dwiputra, R., Moriarty, A., Berghofer, J., Kraetzschmar, G.K.: Design and Development of a Benchmarking Testbed for the Factory of the Future. In: 2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA). pp. 1–7. IEEE (2015)
19. Schwab, K.: The Fourth Industrial Revolution. Penguin UK (2017)
20. Sutcu, Y., Sencar, H.T., Memon, N.: A Secure Biometric Authentication Scheme Based on Robust Hashing. In: Proceedings of the 7th Workshop on Multimedia and Security. pp. 111–116. MM#38;Sec '05, ACM (2005)
21. Tuyls, P., Akkermans, A.H., Kevenaar, T.A., Schrijen, G.J., Bazen, A.M., Veldhuis, R.N.: Practical Biometric Authentication with Template Protection. In: International Conference on Audio-and Video-Based Biometric Person Authentication. pp. 436–446. Springer (2005)
22. Zuehlke, D.: SmartFactory - Towards a Factory-of-Things. Annual Reviews in Control 34(1), 129–138 (2010)