PROFINET Security: A Look on Selected Concepts for Secure Communication in the Automation Domain

Andreas Walz, Karl-Heinz Niemann, Julian Göppert, Kai Fischer, Simon Merklin, Dominik Ziegler, Axel Sikora

Abstract

We provide a brief overview of the cryptographic security extensions for PROFINET, as defined and specified by PROFIBUS & PROFINET International (PI). These come in three hierarchically defined Security Classes, called Security Class 1, 2 and 3. Security Class 1 provides basic security improvements with moderate implementation impact on PROFINET components. Security Classes 2 and 3, in contrast, introduce an integrated cryptographic protection of PROFINET communication. We first highlight and discuss the security features that the PROFINET specification offers for future PROFINET products. Then, as our main focus, we take a closer look at some of the technical challenges that were faced during the conceptualization and design of Security Class 2 and 3 features. In particular, we elaborate on how secure application relations between PROFINET components are established and how a disruption-free availability of a secure communication channel is guaranteed despite the need to refresh cryptographic keys regularly. The authors are members of the PI Working Group CB/PG10 Security.

PROFINET Security: A Look on Selected Concepts for Secure Communication in the Automation Domain

Andreas Walz*, Karl-Heinz Niemann†, Julian Göppert*,
Kai Fischer¶, Simon Merklin‡, Dominik Ziegler§, Axel Sikora*

*Institute of Reliable Embedded Systems and Communication Electronics (ivESK)
Offenburg University of Applied Sciences, Offenburg, Germany
Email: {julian.goeppert, axel.sikora, andreas.walz}@hs-offenburg.de

†Institute for Sensor and Automation Technology
Hannover University of Applied Sciences and Arts, Hannover, Germany
Email: karl-heinz.niemann@hs-hannover.de, ORCID: https://orcid.org/0000-0001-8931-6789

‡Endress + Hauser Digital Solutions, Reinach, Switzerland
Email: simon.merklin@endress.com

§Siemens AG, Graz, Austria
Email: dominik.ziegler@siemens.com

¶Siemens AG, Munich, Germany
Email: kai.fischer@siemens.com

***Abstract*—We provide a brief overview of the cryptographic security extensions for PROFINET, as defined and specified by PROFIBUS & PROFINET International (PI). These come in three hierarchically defined Security Classes, called Security Class 1, 2 and 3. Security Class 1 provides basic security improvements with moderate implementation impact on PROFINET components. Security Classes 2 and 3, in contrast, introduce an integrated cryptographic protection of PROFINET communication. We first highlight and discuss the security features that the PROFINET specification offers for future PROFINET products. Then, as our main focus, we take a closer look at some of the technical challenges that were faced during the conceptualization and design of Security Class 2 and 3 features. In particular, we elaborate on how secure application relations between PROFINET components are established and how a disruption-free availability of a secure communication channel is guaranteed despite the need to refresh cryptographic keys regularly. The authors are members of the PI Working Group CB/PG10 Security.**

***Index Terms*—PROFINET Security, OT security, secure communication.**

I. Introduction

INDUSTRIAL Ethernet-based communication protocols lacked integrated protection mechanisms in the past. This leads to the situation that these protocols can be attacked. Such attacks have been described already many years ago [1], [2] and also in the recent past [3]. The separation of the automation network (called *cell protection*) was used in the past as a counter measure against attacks, but increasingly, concepts are needed that go beyond classical cell protection [4], [5]. This implies that communication in an automation system needs to be upgraded with integrated security mechanisms, at least integrity protection, as for example described in IEC 62443-3-3 [6]. Sometimes confidentiality of sensitive data is additionally demanded. PROFIBUS & PROFINET International (PI) is currently developing cryptographic security extensions for PROFINET, which address the needs of the market as well as the demands made by relevant security standards [7].

In this paper, we briefly overview the main concepts of *PROFINET Security* (see Section II). Most importantly, we then look deeper at some of the technical challenges faced when integrating cryptographic security into a sophisticated realtime communication protocol like PROFINET (see Section III). Note that we do not strive to discuss and evaluate the PROFINET Security extensions as such, nor the performance penalty of cryptographic protection in PROFINET. The first aspect is left as future work, given the lack of implementations of PROFINET Security (see Section IV). Studies on the second aspect can be found in the literature [8]–[12].

II. PROFINET Security: Conceptual Overview

We start by providing a conceptual overview of PROFINET Security as defined and specified by PI [13]–[15]. A discussion of requirements and attack vectors, as well as the extent to which these can be addressed respectively countered by PROFINET Security can be found in a PI white paper [14].

The authors would like to thank all members of the working group CB/PG10 of PROFIBUS & PROFINET International for their contribution to the security concept and to this paper.

DOI of the original publication: 10.1109/INDIN51400.2023.10217985

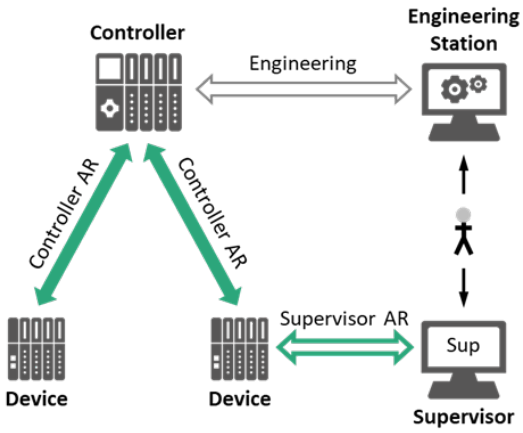

Fig. 1. Illustration of PROFINET Application Relations (ARs). The ARs marked green (Controller AR and Supervisor AR) can be turned into secure ARs if both PROFINET endpoints support at least Security Class 2.

A. PROFINET Security Classes

Three PROFINET Security Classes have been defined (called Security Classes 1, 2 and 3), which classify the security capabilities of PROFINET components and tools.

Compared to classical PROFINET, Security Class 1 introduces improved configuration capabilities for network management (SNMP) and device discovery and configuration (DCP) protocols. Additionally, it introduces and mandates a mechanism for manufacturers of PROFINET components to cryptographically sign their device description (GSD) files. However, Security Class 1 does not introduce any cryptographic mechanisms for protecting PROFINET communication on the network [16].

Security Classes 2 and 3 introduce built-in cryptographic protection of PROFINET communication, allowing endpoints to establish *secure* PROFINET *Application Relations* (ARs, see Figure 1). Secure ARs differ from classical (plain) PROFINET ARs mainly in the following ways:

- The two involved PROFINET endpoints must mutually authenticate themselves using public-key certificates while a secure AR is established.
- *Communication Relations* (CRs) within the secure AR are protected using state-of-the-art symmetric cryptography.
- The invocation of operations and services through a secure AR is enhanced by role-based access control mechanisms.

Security Class 2 offers integrity protection for cyclic and acyclic communication and confidentiality for Record data services. Security Class 3 additionally offers confidentiality for cyclic realtime communication and acyclic alarm messages. Both Security Classes provide dedicated concepts for a *security configuration management* of PROFINET endpoints, including the management of public-key certificates.

PROFINET Security consistently makes use of well-established security standards and technology. For endpoint authentication and cryptographic key establishment, Security Classes 2 and 3 use EAP-TLS, the *Transport Layer Security* (TLS) protocol [17] wrapped into the *Extensible Authentication Protocol* (EAP) protocol [18]. Symmetric cryptographic protection of data transferred within secure ARs is based on instances from the *Authenticated Encryption with Associated Data* family of algorithms [19] (e.g., AES-GCM or ChaCha20-Poly1305 with 256-bit key strength).

B. Discussion and Related Work

Security Class 2 and 3 extensions target cases where an (active or passive) attacker may inject, manipulate, or read messages on the network [20]. Currently, these types of attacks are countered by a zone concept, which strictly limits the access to network zones. The security extensions improve this situation by providing state-of-the-art cryptographic protection for PROFINET messages. However, PROFINET Security is just one building block, and further measures (outside the scope of the PROFINET specification) are necessary for holistic security [15].

For communication security in IP-based networks, (D)TLS is a predominant standard, which different industrial protocols also rely on. MODBUS/TCP and EtherNet/IP can be operated on top of TLS [21], [22]. OPC-UA uses a *Secure Channel* that has similar characteristics as a secure TLS channel [23]. For PROFINET, however, a straight adoption of (D)TLS (or similar protocols) is not possible, e.g., because

1) multiple CRs within a secure AR need to be protected independently and
2) *plain vanilla* TLS is not designed to cope with the tight realtime requirements of PROFINET.

Therefore, PROFINET Security involves some design peculiarities, some of which we illuminate in the following.

III. A Technical Deep-Dive into Particular PROFINET Security Features

Below, we take a deeper look at two particularly interesting technical aspects of PROFINET Security Classes 2 and 3.

- **Secure AR establishment**: how does PROFINET Security integrate endpoint authentication and session key establishment into the AR establishment?
- **Seamless key renewal**: how does PROFINET Security renew cryptographic keys without disrupting the availability of long-lasting secure ARs?

A. Secure AR Establishment

The use of symmetric cryptography requires the involved endpoints to hold and maintain a shared security context. Typically, it contains, among other data, cryptographic keys for message protection and security sequence counters for replay detection. For PROFINET Security, the security context is called *Security Association* (SA). There is a one-to-one relation between SAs and secure ARs. The SA is established while the secure AR is set up, and it is sustained and maintained as long as the AR is alive.

The authentication of PROFINET endpoints is part of the establishment of the SA. EAP-TLS is used to achieve both

in conjunction. It has been chosen for PROFINET Security mainly for three reasons.

- EAP-TLS is one of the most accepted, well-established, and widely-used security protocols for endpoint authentication and session key establishment (for example, EAP-TLS is part of the IEEE 802.1X standard for *Port-Based Network Access Control* [24]).
- EAP-TLS is designed to work in a standardized way on Layer-2 without the need for sockets/IP connections.
- EAP is extensible and allows for non-TLS authentication methods if needed in the future.

PROFINET classically uses a variant of the *Remote Procedure Call* (RPC) protocol or, more recently, the dedicated *Remote Service Interface* (RSI) protocol to implement a service request and response architecture for AR establishment and acyclic record services. With PROFINET Security, the EAP-TLS message exchanges between two PROFINET endpoints are integrated into these RSI/RPC requests/responses.

The secure AR establishment sequence proceeds in roughly four steps, as illustrated in Figure 2. We assume that a PROFINET Controller takes the initiative and wants to establish a secure AR with a PROFINET Device. For other constellations, e.g., a PROFINET Supervisor initiating a secure AR with a Device, the procedure is similar.

1) The Controller sends an initial connect request to the Device via RSI/RPC, indicating that a secure AR shall be established. The request allows for a tentative resource allocation on the Device side. It includes *instructions* which cryptographic algorithms are to be used in the context of the upcoming secure AR. For efficiency reasons the initial connect request also includes the first message of an EAP-TLS handshake (EAP-TLS request). The Device answers with a respective connect response, confirming that it is willing and able to establish the secure AR with the cryptographic algorithms requested by the controller. The connect response also includes the corresponding EAP-TLS response message. Request and response are sent without cryptographic protection.
2) The EAP-TLS protocol flow is continued using dedicated RSI/RPC requests and responses for the EAP-TLS request and response messages, respectively. As a result of a successfully finished EAP-TLS handshake, both sides hold a shared secret, which they use to initialize the cryptographic keys of the SA. At this point, cryptographic protection for any further PROFINET communication occurring within the context of the secure AR becomes effective. Note that the Controller takes the EAP *authenticator* role and, as a result of the EAP-TLS design, the TLS server role, while the Device takes the EAP *supplicant* and the TLS client role.
3) With the cryptographic protection being in effect, the initial connect request/response pair is repeated (this time without including the initial EAP-TLS messages). The repetition allows the Device to retrospectively verify the integrity of the initial connect request and the selection of cryptographic algorithms it included. With a successful confirmation, the SA is fully established.
4) The AR establishment sequence, as classically used by PROFINET, is continued. In contrast to the case of plain ARs, though, the respective RSI/RPC requests and responses for setting up and parametrizing the AR are now cryptographically protected.

B. Seamless Key Renewal

In the course of the secure AR establishment sequence, as described in the previous section, cryptographic protection becomes and stays effective for all PROFINET communication occurring within the secure AR.

However, there is an important conflict that needs to be resolved. On the one hand, PROFINET ARs, whether secure or plain, potentially must last for years while allowing for continual communication without any disruption. On the other hand, the continual use of cryptographic keys is subject to limitations, such that, at some point in time, a renewal of keys is necessary, as explained in the following.

- First, cryptographic message protection generally involves security sequence counters. These counters render individual messages unique, even if their content is identical. This is important, e.g., to allow detecting malicious message replay attacks. These counters naturally have a limited range, but must never overflow within the lifetime of a cryptographic key. This requirement leads to a hard limit on the number of messages that can be protected under a single cryptographic key.
- Second, the security strength of cryptographic protection degrades with extensive usage of a single cryptographic key. This fact leads to a soft limit on the number of messages and the data volume that can be protected under a single cryptographic key [25], [26].

Note that the need to renew cryptographic keys is not an issue specific to PROFINET Security. As explained above, it is generally found in cryptographic systems. However, in the context of PROFINET Security, the matter requires special attention and treatment. In typical office IT settings, key renewal can occur under significantly relaxed requirements. For PROFINET, however, any disruption of ongoing realtime communication is unacceptable and must be excluded by all means. Additionally, PROFINET's cyclic communication potentially leads to large data volumes, requiring frequent key renewals. This disqualifies naïve key renewal approaches, which may imply additional network exchanges or which cannot guarantee the seamless availability of keys.

Another challenge that is relevant for PROFINET Security and that requires consideration are strict timing requirements. PROFINET implementations often rely on dedicated hardware for cyclic realtime communications. The computation of new cryptographic keys, on the other hand, is most likely going to take place in software. That is, the seamless key renewal process potentially needs to take place across hardware/software boundaries. Interactions between hardware and software modules, however, may feature nondeterministic latencies.

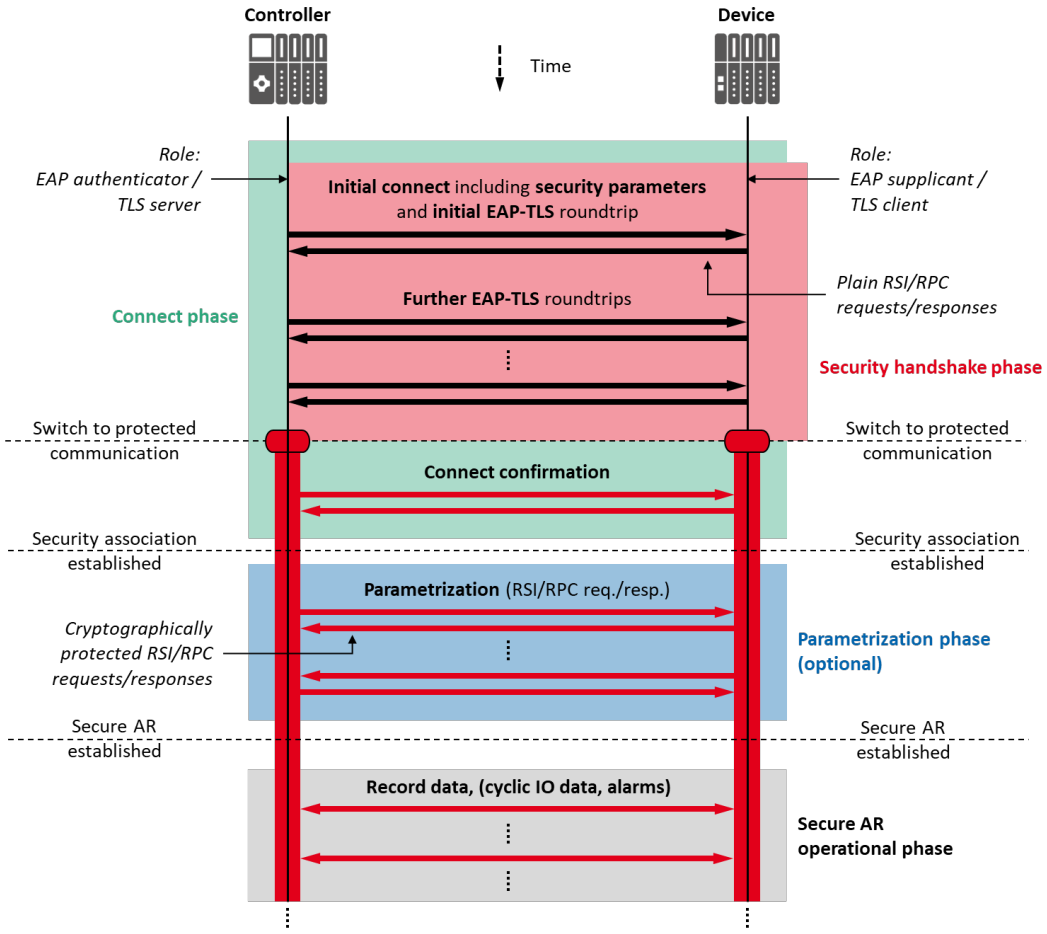

Fig. 2. Illustration of a PROFINET secure AR establishment sequence between a PROFINET Controller and a PROFINET Device. The sequence starts with an initial connect request/response pair sent in plain, where the request includes instructions as to which security parameters (cryptographic algorithms) to use. Additionally, this first roundtrip includes an initial EAP-TLS request/response pair. Thereafter, the EAP-TLS handshake is continued by embedding the corresponding EAP-TLS requests and responses in dedicated RSI/RPC requests and responses, respectively. With the successful termination of the EAP-TLS handshake, cryptographic protection becomes effective. Another connect request/response sequence follows, which serves to verify the integrity of the initial plain connect retrospectively. All following PROFINET communication proceeds as it does in the classical case, but with cryptographic protection being enabled.

Therefore, PROFINET Security uses a careful design to facilitate a seamless renewal of cryptographic keys. It is based on the proposal by Bühler et al. [27], [28]. The design allows a deterministic evolution of cryptographic keys *without* imposing additional network load and *without* imposing strict timing requirements on the key computation procedure. In doing so, it does not risk seamless key availability.

The following key features underlie the design. We distinguish between key *installation* (the key is put in place, but not yet used) and key *activation* (key usage starts).

- The key renewal process is asymmetric. The initiator of a secure AR (most often a PROFINET Controller) is the endpoint that is in charge of deciding when to execute the renewal process. The responder side (most often a PROFINET Device) follows the respective instructions of the initiator side. This allows Controllers, which may need to manage and maintain a large number of secure ARs at the same time, to retain control over the distribution of key renewal load.
- The key renewal is a lock-step procedure with *announcements* and *confirmations*. The initiator of an AR can announce to the other endpoint that a key renewal process needs to be executed soon. This allows either side to perform required preparative actions without the need to adhere to strict time constraints. For example, only after the responder has confirmed a successful key computation, the new key is activated.
- The key renewal process uses generous grace periods. The key renewal happens for all CRs of a secure AR and both communication directions roughly at the same time. However, there is no need to strictly synchronize the activation of the new key across CRs and directions. This allows individual CRs and directions within a single secure AR to lag behind with key activation for some time. As a result, synchronization demands, e.g., between hardware and software modules, are relaxed.

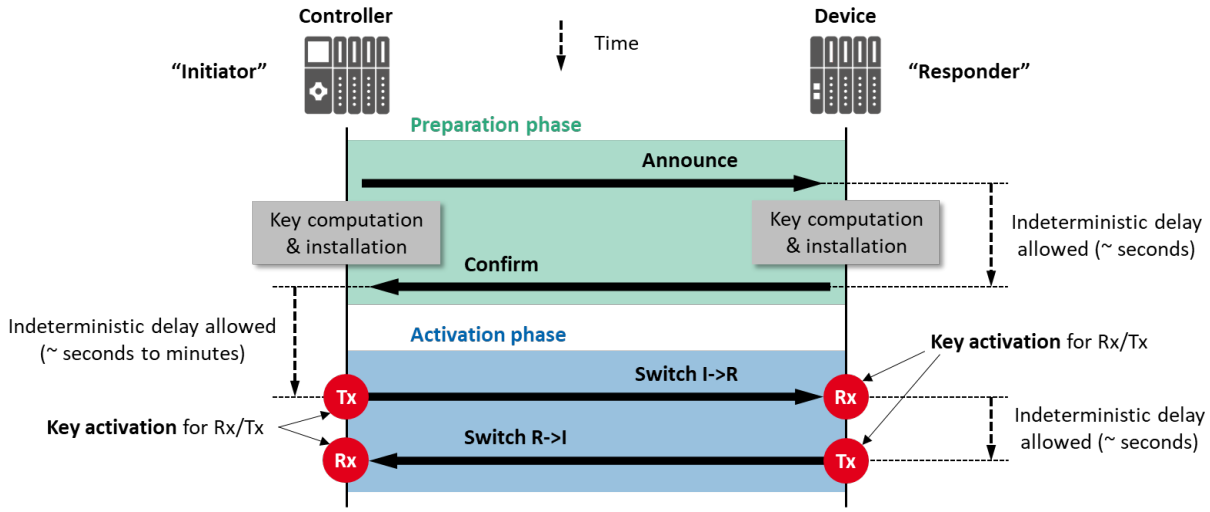

Fig. 3. Illustration of the key renewal procedure underlying PROFINET Security. It includes two phases (preparation and activation phase), with two steps each. During the preparation phase, the two endpoints inform and assure each other of their readiness to use a new cryptographic key. During the activation phase, the new key is put into effect. The process follows a lock-step paradigm, in which each side waits for the other side to complete the previous step. All interaction between initiator and responder are carried in line with protected PROFINET messages that are exchanged between the two anyway.

- All information, which needs to be exchanged between the two endpoints to orchestrate the key renewal process, is carried within regular PROFINET messages that are sent independent of the key renewal procedure. To this end, a dedicated and always-present field is used, which is part of the security metadata that is added to each cryptographically protected PROFINET message. This allows to run the process without any effect on the volume or frequency of network traffic.

The key renewal proceeds in four steps, grouped in two phases, as illustrated in Figure 3.

- **Announcement (preparation phase)**: the initiator of a secure AR announces to the responder that a key renewal process is pending. This causes the responder to schedule the corresponding cryptographic computations. The initiator may have done the same computations before issuing the announcement, or it may do so only after the announcement.
- **Confirmation (preparation phase)**: once the responder has successfully computed and installed the new key, it issues a confirmation to the initiator side. This information provides assurance to the initiator that the responder is indeed ready to activate the new key.
- **Switch Initiator to Responder (activation phase)**: after having successfully computed and installed the new key and after having received the confirmation of the responder, the initiator decides at its own discretion when to active the new key for outgoing messages. On the responder side, the new key is already installed, too. Once the responder receives the first message protected with the new key, it activates the new and deactivates the previous key for incoming messages.
- **Switch Responder to Initiator (activation phase)**: with the reception of the first message protected with the new key, the responder is free to decide at its own discretion when to activate the new key for outgoing messages, too. Once the initiator receives the first message protected with the new key, it activates the new and deactivates the previous key for incoming messages.

After key activation for incoming messages, messages protected with the previous key are not accepted anymore.

The process is complete after the initiator has finally activated the new key for incoming messages.

C. Discussion and Related Work

Various approaches to integrate cryptographic security into PROFINET have been proposed before. Some of them focus just on the cryptographic protection of cyclic realtime CRs [10], [29]. In contrast to the design of PROFINET Security (as sketched within this section), these approaches do not offer a protection of, e.g., device parametrization or other relevant CRs within a PROFINET AR (e.g., record services). Moreover, they do not address the establishment and renewal of the symmetric cryptographic keys required for message protection.

Runde et al. presented a comprehensive approach for integrating cryptographic security into PROFINET [30]. It uses the *Internet Key Exchange* protocol in version 2 (IKEv2) [31] with a custom extension to authenticate endpoints and to establish and renew cryptographic keys. PROFINET Security, in contrast, uses EAP-TLS and tightly embeds it into the AR establishment sequence. The novel and crucial benefit of this design is that it allows to cryptographically protect PROFINET's RSI/RPC exchange sequences *entirely*, *without* requiring additional protocol exchanges to take place outside of or prior to PROFINET's RSI/RPC communication. Also, with its dedicated key renewal process, the design of

PROFINET Security does not require additional network traffic to accomplish a renewal of its (symmetric) cryptographic keys.

IV. Summary, Conclusion and Outlook

We provided an overview of *PROFINET Security*, which allows a cryptographic protection of PROFINET communication (e.g., realtime data, acyclic data, etc.) using well-established security technologies. As our main focus, we shed light on two particularly interesting aspects of the design of PROFINET Security: first, how it integrates endpoint authentication and cryptographic key establishment into the AR establishment sequence and, second, how it seamlessly renews cryptographic keys without imposing additional network traffic.

PROFINET Security still is a young concept. Its features have been integrated into the PROFINET specification by now [13], even though further refinements of the specification can be expected. Unfortunately, implementations of the security extensions are not yet (publicly) available. This strongly limits the possibilities to evaluate the concept in reality. Therefore, a real-world prototyping of the specified concepts in a test bed is among the most important next steps.

Once implementation experience with PROFINET Security has been gained, further work and publications are planned. This includes, for example, a detailed evaluation of its effectiveness and efficiency: does it retain the realtime capabilities of PROFINET in general, and what are the quantitative limitations imposed by PROFINET Security in particular? Furthermore, guidance on the implementation of PROFINET Security is planned to be given, and lessons learned from developing a prototype implementation are planned to be shared.

References

[1] M. Baud and M. Felser, "Profinet io-device emulator based on the man-in-the-middle attack," in *2006 IEEE Conference on Emerging Technologies and Factory Automation*, 2006, pp. 437–440.

[2] J. Akerberg and M. Bjorkman, "Exploring security in profinet io," in *2009 33rd Annual IEEE International Computer Software and Applications Conference*, vol. 1, 2009, pp. 406–412.

[3] S. Mehner and H. König, "No need to marry to change your name! attacking profinet io automation networks using dcp," in *Detection of Intrusions and Malware, and Vulnerability Assessment*, R. Perdisci, C. Maurice, G. Giacinto, and M. Almgren, Eds. Cham: Springer International Publishing, 2019, pp. 396–414.

[4] K.-H. Niemann and M. Hoh, "It security of field devices – contributions to the it security of production plant networks," *atp magazin*, vol. 59, no. 12, 2017.

[5] K.-H. Niemann and S. Merklin, "Ot security requirements for ethernet-apl field devices," *atp magazin*, vol. 64, no. 5, 2022.

[6] International Electrotechnical Commission (IEC), "IEC 62443: Security for industrial automation and control systems part 3-3: System security requirements and security levels," 2013.

[7] ——, "IEC 62443: Security for industrial automation and control systems part 4-2: Technical security requirements for iacs components," 2019.

[8] B. Czybik, S. Hausmann, S. Heiss, and J. Jasperneite, "Performance evaluation of mac algorithms for real-time ethernet communication systems," in *2013 11th IEEE International Conference on Industrial Informatics (INDIN)*, 2013, pp. 676–681.

[9] M. Runde, C. Tebbe, and K.-H. Niemann, "Performance evaluation of an it security layer in real-time communication," in *2013 IEEE 18th Conference on Emerging Technologies Factory Automation (ETFA)*, 2013, pp. 1–4.

[10] T. Müller and H. D. Doran, "PROFINET Real-Time Protection Layer: Performance Analysis of Cryptographic and Protocol Processing Overhead," in *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)*, vol. 1, 2018, pp. 258–265.

[11] M. Skuballa, A. Walz, H. Bühler, and A. Sikora, "Cryptographic protection of cyclic real-time communication in ethernet-based fieldbuses: How much hardware is required?" in *2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2021, pp. 1–7.

[12] S. Hohmann, T. Mueller, and M. Stübs, "Bridge me if you can! evaluating the latency of securing profinet," in *2021 International Conference on Information Networking (ICOIN)*, 2021, pp. 621–626.

[13] PROFIBUS Nutzerorganisation e.V., "Application layer protocol for decentralized periphery technical specification for profinet io: Version 2.4 mu3." [Online]. Available: https://www.profibus.com/download/profinet-specification

[14] ——, "Security Extensions for PROFINET - PI White Paper for PROFINET." [Online]. Available: https://www.profibus.com/download/pi-white-paper-security-extensions-for-profinet/

[15] K.-H. Niemann, A. Walz, and A. Sikora, "Security extensions for profinet: Concepts, status, and prospects," Embedded World Conference 2023.

[16] PROFIBUS Nutzerorganisation e.V., "Security class 1 for profinet-security." [Online]. Available: https://www.profibus.com/download/profinet-security-guideline

[17] E. Rescorla and T. Dierks, "The Transport Layer Security (TLS) Protocol Version 1.2," RFC 5246, Aug. 2008. [Online]. Available: https://www.rfc-editor.org/info/rfc5246

[18] D. Simon, R. Hurst, and D. B. D. Aboba, "The EAP-TLS Authentication Protocol," RFC 5216, Mar. 2008. [Online]. Available: https://www.rfc-editor.org/info/rfc5216

[19] D. McGrew, "An Interface and Algorithms for Authenticated Encryption," RFC 5116, Jan. 2008. [Online]. Available: https://www.rfc-editor.org/info/rfc5116

[20] K.-H. Niemann, "It security extensions for profinet," in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, vol. 1, 2019, pp. 407–412.

[21] MODBUS.org, "MODBUS/TCP Security – Protocol Specification." [Online]. Available: https://modbus.org/docs/MB-TCP-Security-v21_2018-07-24.pdf

[22] ODVA, "Overview Of CIP Security." [Online]. Available: https://www.odva.org/wp-content/uploads/2020/05/PUB00319R1_CIP-Security-At-a-Glance.pdf

[23] OPC Foundation, "Opc 10000-4: Ua part 4: Services." [Online]. Available: https://opcfoundation.org/developer-tools/documents/view/161

[24] "Ieee standard for local and metropolitan area networks–port-based network access control," *IEEE Std 802.1X-2020 (Revision of IEEE Std 802.1X-2010 Incorporating IEEE Std 802.1Xbx-2014 and IEEE Std 802.1Xck-2018)*, pp. 1–289, 2020.

[25] A. Luykx and K. G. Paterson, "Limits on authenticated encryption use in tls," 2016. [Online]. Available: https://www.atul.be/aelimits_2017_08_28.pdf

[26] F. Günther, M. Thomson, and C. A. Wood, "Usage Limits on AEAD Algorithms," Internet Engineering Task Force, Internet-Draft draft-irtf-cfrg-aead-limits-07, May 2023, work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/draft-irtf-cfrg-aead-limits/07/

[27] H. Bühler, A. Walz, and A. Sikora, "A mechanism for seamless cryptographic rekeying in real-time communication systems," in *2021 17th IEEE International Conference on Factory Communication Systems (WFCS)*, 2021, pp. 53–58.

[28] H. Bühler, "Definition, implementation and verification of a fpga-based design for secure profinet rtc communication," Master Thesis, Offenburg, 2020. [Online]. Available: https://opus.hs-offenburg.de/files/4201/20200927_thesis_final.pdf

[29] J. Åkerberg and M. Björkman, "Introducing security modules in profinet io," in *2009 IEEE Conference on Emerging Technologies Factory Automation*, 2009, pp. 1–8.

[30] M. Runde, "Echtzeitfähige protokollerweiterung zum schutz ethernet-basierter automatisierungskomponenten," Ph.D. dissertation, Otto-von-Guericke-Universität Magdeburg, 6 2014.

[31] C. Kaufman, "Internet Key Exchange (IKEv2) Protocol," RFC 4306, Dec. 2005. [Online]. Available: https://www.rfc-editor.org/info/rfc4306