IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 20 December 2023, accepted 2 January 2024, date of publication 8 January 2024, date of current version 22 January 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3350778

TOPICAL REVIEW

Securing Sensing in Supply Chains: Opportunities, Building Blocks, and Designs

JAN PENNEKAMP[1], (Graduate Student Member, IEEE),
FRITZ ALDER[2], (Graduate Student Member, IEEE),
LENNART BADER[3], (Associate Member, IEEE),
GIANLUCA SCOPELLITI[2,4], KLAUS WEHRLE[1], (Member, IEEE),
AND JAN TOBIAS MÜHLBERG[2,5]
[1]Communication and Distributed Systems, RWTH Aachen University, 52074 Aachen, Germany
[2]DistriNet, KU Leuven, 3001 Heverlee, Belgium
[3]Cyber Analysis and Defense, Fraunhofer FKIE, 53177 Bonn, Germany
[4]ER Security Sweden, Ericsson, 16483 Stockholm, Sweden
[5]BEAMS and Cybersecurity Research Center, Université Libre de Bruxelles—École Polytechnique, 1050 Brussels, Belgium

Corresponding author: Lennart Bader (lennart.bader@fkie.fraunhofer.de)

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy-EXC-2023 Internet of Production-390621612. This work was further supported in part by the Research Fund KU Leuven and received funding under EU H2020 MSCA-ITN Action 5GhOSTS, grant agreement no. 814035. The work of Fritz Alder was supported by a grant of the Research Foundation-Flanders (FWO) under grant number 11E5120N.

ABSTRACT Supply chains increasingly develop toward complex networks, both technically in terms of devices and connectivity, and also anthropogenic with a growing number of actors. The lack of mutual trust in such networks results in challenges that are exacerbated by stringent requirements for shipping conditions or quality, and where actors may attempt to reduce costs or cover up incidents. In this paper, we develop and comprehensively study four scenarios that eventually lead to end-to-end-secured sensing in complex IoT-based supply chains with many mutually distrusting actors, while highlighting relevant pitfalls and challenges—details that are still missing in related work. Our designs ensure that sensed data is securely transmitted and stored, and can be verified by all parties. To prove practical feasibility, we evaluate the most elaborate design with regard to performance, cost, deployment, and also trust implications on the basis of prevalent (mis)use cases. Our work enables a notion of secure end-to-end sensing with minimal trust across the system stack, even for complex and opaque supply chain networks.

INDEX TERMS Blockchain technology, reliability, security, trust management, trusted computing, trusted execution environments.

I. INTRODUCTION

Supply chain management involves provisioning, internal and external suppliers, vendors, logistics, bookkeeping, and billing [1], [2]. Depending on the final product, supply chains can be highly complex as they include a copious number of actors and business interests, many interdependent production steps, and potentially high levels of variability and uncertainty [3], [4]. Opaque trust relationships in large-scale distributed digital infrastructures—e.g., sensing and processing systems, complex communication paths, and mobile networks—add to this significant level of complexity.

While corresponding technical research on supply chain management largely and extensively focuses on the *flow* of information [1], advances in Cyber-Physical Systems (CPS) and in the Internet of Things (IoT), especially its *comprehensive sensing capabilities*, highlight the importance of considering the information's *origin* and *acquisition processes*. In this paper, we do not limit ourselves to a specific type of sensor. As a result, the term "sensing" collectively refers to different strains of data acquisition, ranging from simple barcode scanners to image-based video capturing.

The associate editor coordinating the review of this manuscript and approving it for publication was Leandros Maglaras.

Emergent paradigms like the Internet of Production [5] or the Physical Internet [6], e.g., emphasize collaboration between manufacturing companies with strong supply-chain integration across computational infrastructures, involving a "global logistics system that aims to move, handle, store, and transport logistics products in a sustainable and efficient way" [6]. These sensing-based approaches have drastically reshaped business processes, resulted in increasingly automated decision-making [7], [8], [9], and established new application domains, including industrial applications where IoT systems are used to monitor manufacturing processes [10], specifically in the context of intra-corporate deployments [3], [9], [11]. Apart from academia, IoT-oriented companies (e.g., upkeep [12], project44 [13], or roambee [14]) also seize the moment to widely deploy modern sensing technology in supply chains. Fittingly, recent research also investigates the use of such sensing equipment in the context of multi-stakeholder deployments that are characteristic of complex supply chain networks [8], [15], [16]. The corresponding large number of actors opens up important questions around *cyber security, dependability, trust relationships, authenticity, and accountability* to enable robust and resilient supply chains [17], [18].

Addressing the research gap of improving this data reliability is an important and timely challenge [19]. In prior work [17], we argued that end-to-end (E2E) secure means of supply chain monitoring can be implemented using Trusted Execution Environments (TEEs), tightly interwoven with immutable storage such as immutable databases based on Merkle trees [20] or public ledgers [21]. Specifically for perishable supply chains—cold chains in food supply or pharmaceuticals—E2E secure setups seem to introduce acceptable infrastructure costs, enable near real-time monitoring and detection of incidents, and minimize the required trust between the involved parties. This prior work covers two aspects: (1) How to ensure that sensor data can be trusted, especially if multiple stakeholders sense, forward, process, and use the information, and (2) how and when distrusting stakeholders can trust and rely on each others' (historic) sensor data. However, due to the paper's scope, we left some design details and security considerations for future work. Additionally, we only discussed selected deployment aspects, did not feature varying levels of trust relationships, and only briefly covered the performance and security evaluations [17].

A. THIS PAPER

The overall complexity of modern supply chains is naturally reflected in design choices for supply chain monitoring systems, each of which involves different trust relationships and systems costs. The overall impact of design decisions when construing such a (secured) data processing pipeline is not well understood, and the definite applicability of possible solutions is unknown. With this paper, we intend to fill this research gap. Specifically, we design, investigate and compare several supply chain sensing scenarios concerning the previously unexplored security implications, i.e., tamperproofness, authenticity, and accountability, the resulting trust relationships, implementation costs, and performance. In our evaluation, we resort to five common use cases and four realistic misuse cases in supply chains to assess the technical readiness. We further study the performance of all relevant components. As a result, we move the trust in processed data toward real-world requirements and explore the applicability of secure E2E sensing in great detail.

B. CONTRIBUTIONS

Our main contribution is a comprehensive design of four supply chain sensing and monitoring scenarios that involve different approaches to acquiring, processing, and storing sensed data based on the availability of trusted processing elements, management infrastructure for cryptographic credentials, and tamperproof storage solutions. More in detail:

(a) We compile a universally-valid, EPCIS [22]-aligned list of desirable supply-chain-sensing use cases that expresses the respective sensor types, data volumes, sensing frequencies, and latency requirements.

(b) We provide realistic and threatening misuse cases that highlight the potential for actors to deceive others.

(c) We design, investigate, and rate different evolutions for reliable end-to-end secure sensing scenarios that cover all sorts of supply chains. We derive these designs iteratively, which allows for precisely configured real-world deployments (individual needs and attacker models).

(d) We evaluate and discuss the performance, security (misuse cases), and cost implications of the different designs, and place them in the context of real-world use.

We emphasize that any market-ready E2E-secured design should utilize TEE-based sensors and feature a processing infrastructure that eventually persists an immutable trust anchor. Such a design ensures trustworthy processing, long-term availability and authenticity of all data, even in dynamic settings, and with a minimal trusted computing base.

C. OPEN SCIENCE

Our paper's evaluation artifacts are publicly available [23] to ensure reproducibility and reusability.

D. IMPACT BEYOND SUPPLY CHAINS

We believe that our designs, analyses, and evaluation results are useful beyond the scope of supply-chain monitoring but can inform engineering decisions toward secure and trustworthy distributed many-stakeholder systems in the context of IoT, CPS, and for future critical infrastructures. Our prototypic open-source implementation and software development framework could be a valuable case study and starting point for future research to build upon and extend.

II. BACKGROUND: TECHNICAL CONCEPTS

As a foundation for our work, we now give a concise overview of the two most-relevant, technical (security) concepts.

A. TRUSTED COMPUTING

Trusted computing is a set of hardware security mechanisms to shield software on a device from untrusted access [24], [25]. Based on a hardware root of trust, software can isolate itself to become integrity and confidentiality protected from the surrounding untrusted operating system [24]. While different approaches can achieve this isolation, Trusted Execution Environments (TEEs) are a popular method that either provide separate trusted and untrusted environments, like ARM TrustZone [26], or that granularly shield specific memory regions, like Intel SGX [27] or Sancus [28]. In both cases, TEEs typically enable isolated software to additionally *attest* itself to remote stakeholders. This attestation allows to bind an authenticated and encrypted communication channel to a specific component on the trusted computing device, giving the remote stakeholder the guarantee that they can verifiably and securely communicate with specific software on the device, which is useful for mutually distrusting actors. Related work also demonstrates reliable attestation mechanisms between TEEs from different vendors [29], [30].

Concerning IoT and environmental sensing, some TEEs allow the direct (i.e., secure) control of peripherals by trusted software [24], [25]. This feature enables remotely attestable software to provide authentic and integrity-protected measurements that are independently verifiable by remote parties. We refer to them collectively as *trusted sensors*.

B. BLOCKCHAIN TECHNOLOGY

Blockchain technology orthogonally can support the reliable long-term storage of information in settings with mutually distrusting parties. As immutable and distributed append-only ledgers, blockchains utilize cryptography to irrevocably link information-containing *blocks* to form a chain [31]. By appending new blocks, altering or removing older blocks becomes computationally infeasible. Blockchains hence achieve *tamperproofness* and can guarantee the *existence* of data in distributed settings, replacing trusted third parties. These long-term guarantees make blockchains a valuable tool to enhance collaborations in (potentially) low-trust supply chains [32], [33], [34]. Despite its accountability features, scalability issues regarding computational and storage overheads remain open challenges [35], [36] and require careful consideration of the used consensus algorithm and the amount of data to store [37]. A common approach for minimizing both performance and storage overhead is to only persist *fingerprints* of the data on the blockchain [33]. Simultaneously, fingerprints also mitigate common privacy concerns.

Over time, different variants of blockchains have emerged, namely *public* and *permissioned*. They offer different trade-offs regarding accountability, privacy, and scalability: While public blockchains are usually operated publicly without focusing on a single use case and offer a high degree of accountability, they particularly face scalability issues and require special considerations of privacy aspects. Permissioned (or *private*) blockchains usually offer better scalability because they are tailored to specific applications and operate with specialized consensus algorithms. However, their degree of accountability is arguably weaker than with public blockchains. Similarly, implementation details, the chosen consensus algorithm as well as the blockchain's scale, i.e., the number of participating nodes, directly influence the blockchain's practical security and accountability guarantees.

III. SUPPLY CHAIN SENSING AND PROCESSING

Based on interactions with supply chain experts, we now briefly illustrate supply chains, the involved actors, as well as desirable sensing use cases while only presenting details that are relevant for our setting—the sensing in supply chains. Sourcing this overview, we then compile a list of goals for the reliable data collection and processing in supply chains.

A. SCENARIO OVERVIEW

Abstracted, supply chains primarily consist of a physical dimension (the flow of shipments, parcels, products, and paper-based documentation) and a digital one, which covers the exchange of information and data along the supply chain. Most importantly, supply chains are usually composed of multiple stakeholders where, especially over multiple hops, pre- and succeeding companies might not be trusted or even known to data-processing actors. Thus, the digital dimension requires security mechanisms to ensure a trustworthy and reliable flow of information across companies and stakeholders.

1) INVOLVED STAKEHOLDERS

In supply chains, various actors can sense, forward, and process information: (i) *production-related* commodity corporations, manufacturing companies, and machine suppliers; (ii) *logistics-related* shipping companies, customs authorities, and warehousing services; and (iii) *sales-related* distributors, retailers, and customers. Particularly, sales-related stakeholders are interested in the product's origin, i.e., through its history of sensed data, whether it may be due to fair-trade, sustainability, or authenticity needs. Thus, within a supply chain, a multitude of (sensed) data can be demanded and provided by individual stakeholders at the same time.

2) SENSING AND MONITORING

Information on a shipment is critical for today's supply chain management, especially for new paradigms [5], [6]. Broadly, we identify three groups of shipment information: (i) Tracking data, i.e., where the shipment is; (ii) monitoring data, i.e., what condition the shipment is in; and finally, (iii) information on the product itself (e.g., to swiftly adapt production processes). Gathering and handling this data is of utmost importance to the entire supply chain as it allows companies to manage their processes and schedule their operations.

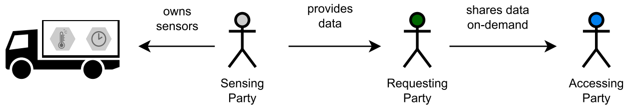

FIGURE 1. **Overview of the logical actors: Sensing parties own sensors and forward data to requesting parties who may share it with accessing parties on-demand for later use (potentially significantly later in time).**

3) RESEARCH GAP

From this brief description, we derive that the flow of information and its distributed sensing within supply chains is challenging and complex. Numerous threats during sensing, forwarding, processing, and (long-term) storage arise due to the large number of actors with potentially non-existing or little trust relationships. Unfortunately, today's supply chains lack the technical means to satisfactorily address them. As a result, many actors still use inefficient processes (e.g., paper-based reports), do not collect desired information, or refrain from trusting and utilizing received (remote) data.

Thus, we identify the need to provide actors with a secure E2E sensing design to allow them to reliably (trustworthy and timely) detect undesirable delivery statuses or environmental conditions of their shipments and other issues, even when untrusted stakeholders reported otherwise. In fact, we even plan for *distrust* between stakeholders, where different stakeholders have no prior reason to trust one another. Long-term availability of data would further improve the actors' chances for accountability and (data) verifiability, e.g., to unequivocally assign blame in case of disputes. Hence, closing this research gap is especially beneficial for issues that are not immediately noticeable, like a temporarily interrupted cold chain.

B. LOGICAL ACTORS IN SUPPLY CHAINS

When analyzing the flow of sensed data in supply chains, we identify three classes of logical actors: sensing, requesting, and accessing parties. We illustrate their relation in Figure 1. Conceptually, each stakeholder can take the roles of multiple logical actors simultaneously.

First, a *sensing party* owns and deploys the sensors in use. It also makes sure to provide the sensed information to the requesting party. The sensing party is usually a shipment provider during transit, but for shipments with sensitive, expensive, or fragile cargo, a customer might also request the inclusion of its own sensors. Likewise, warehousing departments can act as sensing parties when utilizing smart readers to process incoming or outgoing shipments.

Second, *requesting parties* are the intended, original recipients of sensed data, e.g., customers requesting details on their purchase and the transit of their goods. The requesting party may also be the sensing party (for documentation and benchmarking purposes). Still, sensed information is usually relevant to only one party. However, if shipment providers group cargo by different customers in a single container, all customers might be interested, e.g., in maintaining the cold chain. Focusing on potential data availability needs, the requesting party is responsible for the long-term storage.

Third, additional *accessing parties* might be interested in the sensed information at a later point in time. Accessing parties can be virtually any stakeholders that are concerned with the supply chain, from production companies, over suppliers, retailers, and governmental agencies, to end customers. In this case, the sensed information is originally shared by the requesting party, and if no direct business relationship between the accessing and requesting party exists, the information must pass multiple hops. For example, following an accident, data might only become relevant after years.

C. TYPICAL SENSING IN SUPPLY CHAINS

Based on the electronic product code information services (EPCIS) industry standard [22], [38], we can identify a number of relevant sensing use cases in supply chains. The corresponding EPCIS data model captures the dimensions of what, when, where, and why [22], the latter being irrelevant in our scenario. The recent successor, which is intended for state-of-the-art supply chain data interoperability, further includes the dimension "how" [39], matching the research gap of providing companies with reliable (sensed) data.

1) OVERVIEW OF RELEVANT USE CASES

When looking at different sensing applications within supply chains, we identify five general use cases with increasing (technical) complexity and processing requirements. They are not limited to a single granularity (e.g., shipments only). Instead, they can either be applied on a shipment, parcel, or product level according to the specific use case needs. By default, each measurement includes the dimensions *what* (i.e., the focus of the measurement) and *when* (i.e., timing information). Thus, we focus on *where* and *how* instead.

a: STATUS TRACKING

To track the status of a physical flow (along a supply chain), requesting parties are interested in corresponding status changes. For example, when handling a parcel, such as moving it from a container to a warehouse or changing the mode of transportation (e.g., from truck to aircraft), this information, as well as its location (*where*), must be recorded. This tracking can be achieved using stationary RFID readers or BLE beacons (*how*). To the general public, this use case is well-known from the last-mile tracking of consumer parcels.

b: LOCATION TRACKING

If requesting parties also want to know the approximate locations of products, they demand periodic or real-time updates of the respective locations. Thus, location sensors (*how*) must reliably sense this information (*where*). In the context of consumer parcels, this sensed location data is nowadays frequently available as part of last-mile deliveries.

TABLE 1. Technical overview and equipment of our use cases.

Use Case	Sensor	Payload	Measure. Frequency	Time Criticality
Status Track.	Smart Reader	<1–<100 KiB	Triggered	Minutes
Location Track.	e.g., GPS	<1 KiB	<1 records/min	Minutes
Integrity Mon.	Smart Lock	<1 KiB	Triggered	Hours
Condition Mon.	Various	<1–<10 KiB	<6 records/min	Hours
Visual Mon.	Camera	>10 KiB	Variable	Variable

c: INTEGRITY MONITORING

Moreover, companies might also be interested in the (physical) integrity of their shipments (*where*). For example, pharmaceuticals, detailed documentation is even required by law [40]. Likewise, potential violations regarding the integrity of shipments cannot only result in monetary damages but also in harm to humans (e.g., food poisoning). Thus, precisely capturing such data and maintaining access logs are important aspects. To this end, sensing parties can deploy secure (smart) locks and other surveillance sensors (*how*).

d: CONDITION MONITORING

Extending this previous use case to a continual or real-time yet reliable monitoring (*where*) can be equally relevant. Most prominently, inspecting compliance with the temperature requirements is essential, e.g., to identify spoiled goods in cold chains. Likewise, manufacturers of sensitive products might define constraints for shipment environments. Thus, deployed sensors must reliably provide this information (*how*).

e: VISUAL MONITORING

When considering very valuable products or livestock in transit, video-based monitoring (*where* and *how*) constitutes another use case. While the corresponding image or video feed might only be transmitted after specific triggers, e.g., opening after unlocking a smart lock or when exceeding a specific noise threshold, the exact needs of the involved stakeholders vary significantly depending on the setting.

In real-world deployments, stakeholders commonly rely on a combination of these use cases. Depending on the exact setting, several types of sensors with different densities must be deployed: Location, temperature, humidity, air pressure (altitude), light, shock (impact), acceleration, tilt, or weight sensors, as well as smart locks and scanners (readers).

2) TECHNICAL PERSPECTIVE

In light of the associated computational burden of our use cases, we also consider payload sizes and sensing frequencies (i.e., the processing bandwidth). Latency is of interest to guarantee a timely handling of sensed status or condition changes. In Table 1, we provide an overview of these aspects.

Overall, most use cases have moderate needs when sensing relevant information. The exact payload size depends on the sensor in use, as well as the size of added context information. For the first use case, status tracking, payload sizes can range from 96 bit for the most common type of RFID tags, up to 100 KiB or more for specialized RFID tags with extended user data [41]. Similarly, location tracking and integrity monitoring can be realized in less than 1 KiB, most of the time. For condition monitoring, different sensor types introduce varying payload sizes, with simple measurements covering several bytes. In contrast, visual imagery monitoring or video feeds may lead to more excessive needs (far greater than tens of kilobytes), depending on the required image quality and resolution, as well as available compression methods.

The sensing frequency for condition monitoring can be comparably high (i.e., several measurements per minute). However, the respective latency requirements are usually not demanding because the condition monitoring can often not be acted upon immediately during the transit of a shipment but is instead intended to be an authoritative reference. Essentially, while the information gathered by status and location tracking may be needed within minutes of gathering, integrity and condition monitoring only produce data that has to become available within hours since the data gathering. Data aggregation at the source could help to lower the amount of data to transmit, especially if only data outliers are of interest to requesting parties. Thus, the continual transmission of large payloads is unlikely in most real-world settings.

D. MISUSE CASES: CREDIBILITY THREATS

With multiple supply chain actors, we need to consider several misuse cases in real-world settings that may deceive the requesting or an accessing party, irrespective of their nature, i.e., whether they constitute malicious activity [18], honest mistakes, or carelessness. In this context, we derive that all misuse cases build on one or multiple of the following actions.

1) DATA TAMPERING

Parties might have an incentive to directly manipulate data during transmission for various reasons, e.g., shipment providers trying to cover up shipment treatment deficiencies. This desire may especially emerge in case of accidents that should not reflect negatively on a shipping company or once requesting parties try to deceive accessing parties.

a: DATA HIDING

If direct data tampering is not possible, simply hiding the existence of data or of a specific range of data may be equally desirable. The lack of data may not be surprising to the victim and could easily be blamed on unreliable technology or environmental events such as power outages.

b: DATA INJECTION

A malicious party could attempt to insert forged information, i.e., data that originates from unauthentic and unrelated sources or is made up entirely. Similar to the previous misuse cases, such actions could be useful to deceive actors and convince them to accept the shipment conditions.

E. DESIGN GOALS FOR SECURE E2E SENSING

Based on these threats (misuse cases) and in line with prior work [17], we now compile five pressing (technical) design goals for end-to-end-secured sensing (in supply chains).

G1: Tamperproofness Sensor data must be verifiably untampered when being assessed by the requesting or an accessing party at any point in time. This property is vital to ensure that sensor data can be relied upon by all parties.

G2: Authenticity The design must ensure that sensed data verifiably originates from authentic sensors. Thereby, malicious actors are prevented from (retroactively) forging data. This goal covers both sensing and requesting parties.

G3: Completeness To truly enable E2E-secured sensing, recipients (i.e., the requesting and all accessing parties) must be able to verify that the data they receive is complete. Consequently, other parties must be prevented from removing or withholding any sensed data as well as the existence of sensors. Apart from checking for data completeness from a single sensor, recipients need to confirm that the measurements of all relevant (i.e., deployed and expected) sensors are available.

G4: Affordability Given the overhead of any technical solution, both the (one-time) costs for additional hardware or hardware upgrades and the associated operating costs should be kept to a minimum. Consequently, new designs should be careful to (i) avoid performing computation-intensive tasks and (ii) not introduce excessive duty cycles during (regular) operation. Otherwise, real-world deployments are unrealistic on low-cost IoT devices, preventing widespread adoption.

G5: Latency Agnosticism Any solution must be agnostic to any network latencies or network disruptions (offline periods) experienced by the sensing nodes while generally supporting frequent live updates to infrequent batch uploads.

These goals capture the actors' reliability and security needs. Fulfilling them would greatly improve the basis for the decision-making of all involved stakeholders in a supply chain. However, we still need to consider all relevant pitfalls.

IV. EVOLVING RELIABLE SENSING CONCEPTS

To conflate potentially-diverging business interests of several stakeholders and the demand for increased transparency of shipment status, integrity, and product properties, specifically tailored yet flexible concepts are needed. Here, we propose our design that addresses the challenge of realizing secure E2E sensing in modern supply chains. Considering widely differing supply chain instances with differing needs, we derive our design based on a four-layered scenario model, which we illustrate in Figure 2. Our work bases on remote attestation and authenticated communication. As a high-level overview, we summarize the implications of each design stage on trust, security, and deployment aspects in Table 2.

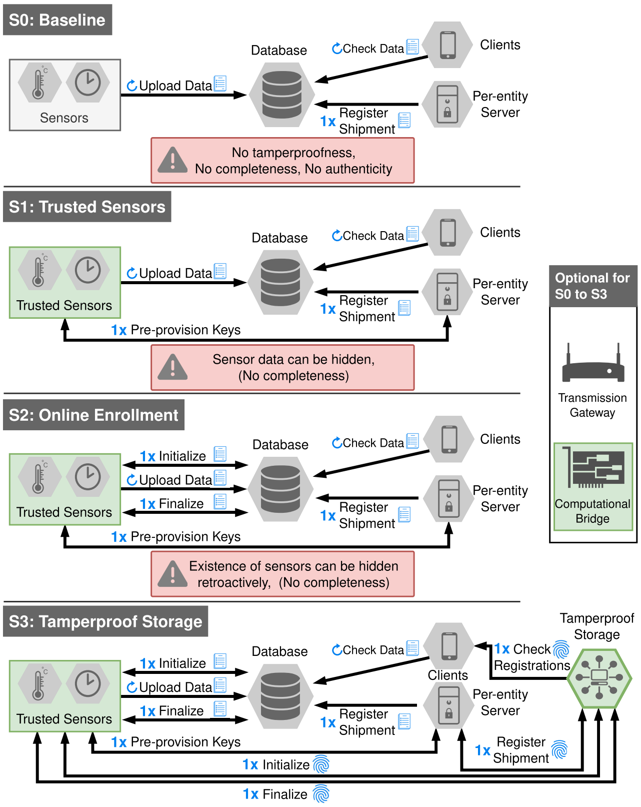

FIGURE 2. Comparison of supply chain sensing scenarios with differing security guarantees. The baseline (S0) does not provide trusted sensing. S1 introduces trusted sensors but lacks data completeness. S2's online phase addresses this issue but lacks sensor completeness. Finally, S3 includes a tamperproof storage for fingerprints to ensure completeness.

The core attacker model that our design needs to protect against stems from the amalgamation of actors and their technical capabilities described in Section III-B as well as the misuse cases described in Section III-D. As such, the system needs to protect against powerful attackers that have strong capabilities akin to the Dolev-Yao model [42] and can have full control over local execution as well as the capability to control network messages. These attackers are only limited by protocols that use established cryptography. On sensors as well as on cloud resources, these capabilities express that attackers are able to locally compromise the entire untrusted execution context except for hardware security mechanisms such as TEEs. When desired, these attackers can also completely isolate devices, drop specific messages, or replay old messages. Where external services and reputation-based stakeholders are involved, we assume a slightly less powerful threat model based on the malicious-but-cautious attacker model [43] due to the fact that cloud or database providers might collaborate with stakeholders but will be limited by actions that do not compromise their reputation.

In the following, we first introduce a traditional baseline sensing scenario in Section IV-A. In Section IV-B, we then

TABLE 2. Comparison of the key design aspects in terms of security and deployment considerations for the scenarios in Figure 2.

	S0: Baseline	S1: Trusted Sensors	S2: Online Enrollment	S3: Tamperproof Storage
Design guarantees that…				
sensor data is not altered	–	✓	✓	✓
sensor data is not deleted	–	–	✓	✓
sensors cannot be hidden	–	–	–	✓
Deployment and Cost				
Communication latency	Configurable	Configurable	Configurable	Configurable
Sensor communication	Offline possible	Offline possible	Online phase required	Online phase required
Storage	Simple database (DB)	Simple DB	Simple DB	Simple DB plus tamperproof storage
Key management	–	Sensor keys	S1 + PKI for sensor receipts in online phase	Same as S2
Additional hardware costs	–	Trusted sensors	S1 + uplink hardware	Same as S2
Additional operational costs	–	–	Network uplink	S2 + tamperproof storage

outline and evolve our design decisions to secure this supply chain. Subsequently, in Section IV-C, we briefly introduce optional components, and, in Section IV-D, we then elaborate on the technical details of the distilled, most-secure scenario. Finally, we discuss deployment challenges in Section IV-E. We defer a discussion on the security and limitations of secure E2E sensing to Section VI.

A. BASELINE SCENARIO: (INSECURE) STATUS QUO

Figure 2 illustrates a baseline scenario, which we denote as S0. Divided into arbitrarily many sensors and three remote stakeholders, the scenario centers around a storage database that is written to by sensing parties, e.g., shipment providers, to register shipments and their involved sensors. Each of these sensing parties is assumed to have its own server infrastructure, denoted as a per-entity server. The database may be maintained by an external party or by a shipment provider and should use, e.g., TLS certificates for authentication. Sensors registered to shipments regularly upload data to the database, which allows authenticated clients to retrieve the data and compare it to their expectations. In all discussed scenarios, clients are seen as an entity that accesses and verifies data, and can, as such, belong to any requesting or accessing party. At this level of abstraction, we focus on the trust a client has toward the other entities, and work toward an environment where the client has to maintain minimal trust in them.

Table 2 lists the guarantees given by the respective designs. The baseline scenario does not guarantee to clients that data has not been altered, deleted, or hidden. The overview shows that this, in practice, well-established baseline scenario only works in supply chains with already existing trust relationships. By being focused on cost-efficiency (G4), only a minimum of financial investments for sensor hardware and standard cloud computing costs is necessary.

However, this scenario is not suited for low-trust environments or for ensuring information integrity (G1–G3). Thus, in the following, we iterate on these trust assumptions to reach a situation where the remaining assumptions are either unavoidable or can be accommodated by external means outside of the system design. Since such a reduction in trust necessarily comes with an increased burden on the client to verify a shipment's validity, Table 3 keeps track of the steps a client needs to take. In S0, simple validation of the sensor data for acceptable parameters suffices.

TABLE 3. Steps to verify the validity in each sensing scenario.

Scen.	Validation Steps (additive)
S0	Perform the following steps for each sensor in the shipment: 1. Check all sensor data for acceptable parameters
S1	2. Check integrity of all sensor data (through TEE attestation)
S2	3. Check the existence of start/end points of sensing periods 4. Check for data/measurement gaps in the sensing period
S3	5. Verify data completeness with tamperproof storage

B. EVOLUTION TOWARD TRUSTWORTHY SENSING

As a first measure to guarantee data authenticity (G2), i.e., to ensure a verifiable information origin and prevent tampering with submitted data (G1), we extend the base scenario with S1 to rely on *trusted sensors*. Attestation, as well as encryption and authentication of all communication, guarantee that involved stakeholders and external actors cannot tamper with data. In both S0 and S1, sensor measurements can be cached offline (locally) until the shipment is completed, and can then be uploaded to the database as a whole dataset. Thus, the only effective change from S0 to S1 is the added hardware cost of employing trusted sensors, the added key management on the side of the sensing party, and the clients' added computational effort to also verify the integrity and authenticity of the data, ensuring that it originates from and is signed by the trusted sensors involved in the shipment. Nonetheless, without further measures, malicious stakeholders can still hide measurements (violating G3), either selectively or also a whole range of data starting at an arbitrary moment during shipment (*data hiding*).

To overcome these issues, the next scenario, S2, includes *online phases* at the beginning and the end of each shipment. During the first online phase, a sensor initializes the shipment by registering itself in the database to confirm its existence and relevance for the shipment. As soon as the shipment is completed, sensors finalize the shipment during the second online phase, indicating the number of measurements of this sensor during the shipment. While the initialization of the shipment signals a clear start of the shipment for future validation, the finalization ensures that no individual measurements are hidden, strengthening data completeness (G3). By involving the trusted sensors in an online phase, clients can verify that no data was lost in the database before sensing started, and that the sensing terminated only upon the final message in the database that also contains the number of measurements during this period. This improvement comes with the added requirement of a live network uplink during the start and end of a sensing period, in addition to the overhead of maintaining an infrastructure that allows the database and sensors to communicate directly with each other. Additionally, the start and end mechanisms have to be actively triggered from the outside, which can happen automatically when a sensor is turned on or through a message from a control server, i.e., as part of a hand-over procedure during shipping. We discuss all associated deployment considerations in Section IV-E.

While S2 already greatly reduces the trust assumptions by the client, it still does not entirely prevent data hiding by removing all data of a sensor. A maliciously acting stakeholder could attempt to retroactively delete all records of a specific sensing from the database before the client accesses this information to filter out sensors with potentially encumbering measurements. Although this attempt might assume a high degree of criminal energy, S3 takes this threat into account. Instead of solely relying on a potentially manipulable database, the respective fingerprints for shipment initialization and finalization are additionally stored on a tamperproof storage (e.g., a permissioned blockchain [33]; we measure the performance of Quorum—a permissioned blockchain—in our evaluation, cf. Section V-A3). To enable a strict ordering of events, the shipment provider is required to also store a fingerprint of the shipment registration on the same tamperproof storage prior to the first sensor initialization. Thereby, we prevent retroactive hiding of sensors and ensure data inclusion for all client verifications. Although primarily relevant for ensuring data completeness (G3), the tamperproof storage further strengthens tamperproofness (G1) and authenticity (G2) in the presence of criminally acting parties. Since the tamperproof storage only stores fingerprints in the form of cryptographic hashes, its usage neither reduces privacy nor scalability. Next, we detail the technical aspects, the information flow, and the respective implications on data trustworthiness and security of our final design.

C. OPTIONAL COMPONENTS IN E2E SENSING

In addition to the main components of our designs (sensors, per-entity server, and database), and for S3, also a tamperproof storage, our design can further be extended with two optional and use-case-specific components.

A **transmission gateway** can be located in proximity of the (trusted) sensors to serve as a *cryptographically-passive* on-path relay that buffers data or simply serves as a network hub. It can be untrusted and does not require any trusted computing hardware. Overall, it can relieve (lightweight) sensors from the overhead of (i) supporting and managing (wireless) communication or (ii) accounting for sufficient buffer sizes (e.g., in use cases with longer offline periods).

In contrast, a **computational bridge** is equipped with a TEE to allow for *cryptographically-attested* on-path manipulations of the sensed data, e.g., to filter or aggregate measurements. Hence, they can also relieve lightweight sensors from complex (pre-)processing tasks. While the resulting computing requirements entail higher deployment costs in practice, computational bridges allow sensing actors to *reliably* reduce the amount of data that needs to be forwarded, stored, and processed without violating any of the introduced end-to-end security guarantees of our work.

D. SECURE E2E SENSING: TECHNICAL GUARANTEES

S3 additionally stores the shipment registration as a fingerprint (cryptographic hash) in the tamperproof storage for two reasons. First, it creates a non-refutable link of used sensors to the shipment. Since the entry is signed by the shipment provider and integrity-protected by the storage, the sensors cannot be disassociated from the shipment anymore. Second, the entry in the storage serves, via relative ordering, as a timing marker for the sensors themselves when they register. Without this ordering, associating sensor data and shipment registration across different types of storage can be difficult. In this context, any sensor activation succeeding a shipment registration that includes this sensor is attached to the shipment, in contrast to requiring time keeping for every sensor. Hence, the long-term reliability of sensed data also depends on the security guarantees of the selected tamperproof storage. When opting for a blockchain-based ledger in S3, the underlying consensus algorithm and the type of blockchain, i.e., a permissioned or permissionless variant, especially influence these security guarantees (cf. Section II-B).

Once a shipment is registered, sensors can be activated and start with their online phase to initialize the sensing. This process again serves two purposes: First, the sensor uploads and verifies the upload of a non-refutable and signed statement that it is now active and can start sensing. This prevents retroactive claims by the shipment provider that the sensor was never active and thus never produced any data. Second, the sensor can provide additional metadata in this initialization message, such as starting conditions or even relative IDs that it uses as start markers in its sensing data.

Due to these purposes, the sensor requires a response from the storage that its entry was included in the database, which requires local verification before the sensor starts operating. In Section IV-E, we discuss how a check of this inclusion proof impacts the sensors' performance requirements.

After initializing the shipment and securing it in the tamperproof storage, the sensors start their operation and upload sensor data to the database, using an internal counter to keep track of the number of measurements made. By design, every sensor data that is stored in the tamperproof storage is also signed to prevent data integrity attacks during data upload. The final measurement counter is then used in the finalization of the shipment to announce and persist it reliably. The last step is again part of an online phase to ensure that the final measurement is also captured in the fingerprint that will be persisted in the tamperproof storage, i.e., the sensing has been terminated gracefully and covers all data.

As a result, clients can (i) retrieve the shipment registration for metadata, (ii) retrieve data of each sensor, and (iii) verify the authenticity and integrity of all received data to ensure it originated from this sensor. Figure 4 in the appendix illustrates the sequence diagram of this verification process.

E. CRUCIAL DEPLOYMENT CHALLENGES

Across these scenarios, various deployment challenges arise: We now discuss the associated key management, nuances for existing trust relationships with the storage provider, checking of inclusion proofs, and the support for optional components.

1) KEY MANAGEMENT

For all scenarios using a trusted sensor (i.e., all but S0), the deployed sensors have to directly communicate with one or multiple storages. To do so securely, sensors and storage have to authenticate each other, which is usually realized through some form of multi-level public key infrastructure (PKI). Sensing parties are expected to pre-deploy devices with keys that exist within this PKI, which can be used by the storage to authenticate the sensor.

Lightweight architectures (e.g., Sancus [28]) may not be able to utilize PKI cryptography, i.e., they only support symmetric key operations. This constraint, however, is simply a deployment concern, as sensing parties can also distribute symmetric keys that are rotated after each shipment, for example, by communicating them to the storage for a specific shipment. Alternatively, a *computational bridge* (cf. Section IV-C) in the form of a more powerful TEE can serve as an on-path processing node between the (trusted) sensor and the database. Since such bridges may also support lightweight sensors by performing the inclusion proof checking, our prototype implementation utilizes one computational bridge to accurately assess the performance impact of such a design.

2) TRUST IN THE STORAGE PROVIDER

We envision an immutable ledger for the tamperproof storage in S3. However, its exact realization can vary based on the considered trust model. For example, if a single trusted entity exists, they could maintain a singular trusted database that also serves as tamperproof storage. As such, our design S3 may, in some situations, be equivalent to S2 if the storage database can be seen as a trustworthy alternative.

Related work provides certain database functionality from inside a TEE [44], [45], which could be another method to establish trust in the database, removing the need for a dedicated tamperproof storage. If the provider running this service can then be verifiably prevented from performing rollback attacks on the data, these concepts are a viable alternative to an immutable ledger and make a TEE-backed storage practically tamperproof.

3) INCLUSION PROOF

S3 places additional work on the sensors to check and verify the inclusion proof received by the tamperproof storage. Depending on the nature and time constraints of this check, it can introduce non-negligible overhead for lightweight sensors. Additionally, depending on the used tamperproof storage, the online part of the communication with the sensor may suffer from undesirable delays, burdening the sensing party.

We see two approaches to deal with these challenges. First, having an online phase immediately before sensing is not necessarily essential as long as the sensor performs the verification at some point. It could start sensing in the meantime. Hence, a delay in communication or computation may be acceptable until the inclusion proof has been verified. The sensor could already sense and upload data to the storage while the inclusion check is running in the background, which would allow to tentatively start a shipment.

If such a time overhead is infeasible, e.g., for constrained devices or short shipment durations, a TEE-based computational bridge between the sensor and the tamperproof storage could be deployed instead, as explained before (cf. Section IV-C). The sensor would directly communicate with the bridge's powerful TEE and receive the confirmation (guarantee) that the inclusion proof has been verified on the tamperproof storage—without any unacceptable delay.

4) GATEWAY TO BUFFER DATA

While we have already discussed a computational bridge, lightweight sensors may additionally benefit from a passive on-path *transmission gateway* (cf. Section IV-C) that buffers data or serves as a network hub (as also suggested by our prior work [17]). Particularly, long-lasting shipments without permanent network connectivity may exceed the storage limitations of constrained sensor nodes, requiring a dedicated gateway between the (trusted) sensors and the database that receives all data from nearby sensors and uploads their data once network connectivity is restored.

V. EVALUATION OF SECURE E2E SENSING

The discussed scenarios each introduce stronger technical sensing and processing guarantees. S1 adds trusted sensors to the baseline scenario to achieve tamperproofness and authenticity (G1-G2), and is improved upon by S2 and S3, which also tackle the issue of data and sensor hiding (G3). In the following, we primarily evaluate our designs with regard to performance, deployment, and cost considerations.

A. PERFORMANCE EVALUATION

Evaluating designs that focus on end-to-end-secured sensing and data processing boils down to the performance of the individual components. Accordingly, we now discuss the performance capabilities and requirements on the two core computational pillars of the discussed designs: (i) The sensing and processing equipment and (ii) the tamperproof storage, here, a permissioned blockchain. All reported performance evaluations are part of our artifacts.

1) SENSING AND PROCESSING EQUIPMENT

In real-world deployments, the selected sensing equipment—microcontrollers, storage, and peripherals—mostly depends on the expected throughput of data and the complexity of the processing to be performed on the sensor node. Specifically, we distinguish between extremely lightweight 16-bit sensing equipment, such as Sancus processors [28] running at 8 MHz, and substantially more powerful 32-bit nodes based on, e.g., ARM Cortex with TrustZone.

To demonstrate the feasibility of our designs, we created a Sancus-based prototype implementation of S3. We show that even under these limiting resource constraints, sensors can address the requirements of S3 and easily handle, i.e., sense and process, most payloads, as compiled in Table 1 (except for video mon.). As we have discussed in Section IV-E, very lightweight sensors can utilize computational bridges to perform computationally intensive operations.

For this reason, the sensors in our prototype utilize an Intel SGX enclave that directly communicates with the Sancus enclaves, and which further serves as a computational bridge between the sensor and the cloud components. In our testbed, we thus simulate the technical requirements of S3 over 100 runs by attaching a XuLA StickIt! [46] board running Sancus 2.1 [47] to an Intel Core i3-7100U running Intel SGX on Ubuntu 22.04.1 LTS, with 16 GB of RAM. Since network conditions are highly use case-specific, this testbed setup abstracts away network delay. Moreover, we do not fix a delay introduced for communicating with the tamperproof storage but perform all computations that would be required for real-world deployments. These operations include the computation of fingerprints for the tamperproof storage (cf. Section V-A3) and signing the data to be sent to the database with a pre-deployed public key certificate. Moreover, the Sancus and Intel SGX enclaves are mutually attested and all on-path communication confidential.

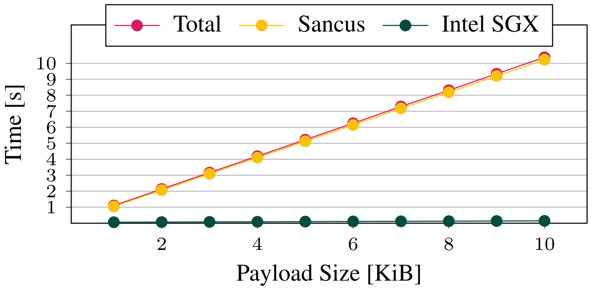

FIGURE 3. Computation and transmission times of data from the trusted sensor to the computational bridge for varying payload sizes. The "Total" and "Sancus" lines are almost overlapping, with nearly all computation time logged at Sancus side.

Thus, this evaluation focuses on the conceptual performance of both hardware components to estimate the feasibility of deploying the most lightweight sensors available.

Starting and finalizing the sensing introduces a computational overhead of 130 ms; 83 ms of this time are consumed by the sensor, and 46 ms are spent by the computational bridge. These numbers exclude the deployment-specific time to verify data inclusion in the tamperproof storage but include all necessary steps to upload the data. In Figure 3, we further detail the individual runtimes of the sensor and computational bridge for common payloads in supply chains between 1 and 10 KiB. The calculated 99 % confidence intervals illustrate only minor deviations over all runs, with a maximum of 138 ms for Sancus and 12 ms for Intel SGX. Each run is triggered externally, simulating a manual sensing request to the sensor. After sensing and encrypting the data, the sensor sends it to the bridge, which re-encrypts the data and signs the fingerprint with pre-deployed PKI keys. This setup allows us to evaluate the longest end-to-end timings and, as such, also serves as an over-approximation of any sensing that could periodically be triggered by the sensor itself, i.e., which is not triggered by an external component or party.

Our evaluation shows that even payloads of up to 10 KiB can be sent (and processed) by sensors roughly 6 times a minute, where the sensor makes up for the majority of computation time and the Intel SGX-based bridge only takes between 55 and 150 ms. Only large payloads, such as high-resolution visual monitoring or specialized RFID tags that contain large unique identifiers of up to 100 KiB (Table 1), are infeasible to be processed on lightweight sensors. In these cases, more powerful microcontrollers should be considered for operation, which may lead to increased deployment costs.

2) OPTIONAL GATEWAY EQUIPMENT

In Section IV-E, we have discussed the use of a cryptographically passive transmission gateway between the deployed sensor and any cloud infrastructure. Depending on the use case, these gateways either purely serve as a networking hub or also cache data during offline periods. In the first case, the gateway simply requires minor storage for the cache. In the second case, the gateway only needs to buffer data until a network connection is restored, which may take anything from minutes to weeks in the case of long-distance shipments. However, for both cases, the observed performance is defined only by the number of sensors, the data size, and the sampling rate of attached peripherals. Considering the previously discussed use cases, each use case would only require a data storage ranging from $1\,\mathrm{KiB} \cdot 60 \cdot 24 = 1440\,\mathrm{KiB} \approx 2\,\mathrm{MiB}$ per day per sensor for small use cases like location tracking and up to $10\,\mathrm{KiB} \cdot 6 \cdot 60 \cdot 24 = 86\,400\,\mathrm{KiB} \approx 100\,\mathrm{MiB}$ per day per sensor for more extensive use cases. With visual monitoring, the data requirements can become arbitrarily large, which would have to be addressed per use case by the gateway.

Overall, gateways can be scaled to serve multiple sensors at a time, leading to latency agnosticism (G5). Thus, we exclude passive networking and storage gateways from our performance evaluation, as their computational requirements are minuscule, and their deployment costs mostly depend on the necessary buffer storage and used communication medium.

3) DATABASE & TAMPERPROOF STORAGE

Considering potential performance bottlenecks after the sensing, we ascertain that all relevant components allow for horizontal scalability: First, despite conceptually being a single entity, the per-entity server (cf. Figure 2) can be implemented by arbitrarily many servers to match the required performance. Second and similarly, each actor can utilize multiple clients for data verification depending on their specific demands. Third, database systems supporting vertical and horizontal scalability are well-established and openly available [33], [48]. Thus, all components allow for horizontal scalability and do not constitute a performance bottleneck.

Given that all information is persisted in a conventional database, storing fingerprints and signatures of said information is sufficient to ensure tamperproofness. Hence, the tamperproof storage solely serves for information integrity and consistency purposes in our design. Since the tamperproof storage can be implemented as a blockchain (cf. Section VI-A), we evaluate the performance of a (private) Quorum blockchain [49] with a proof-of-authority consensus [50]. To this end, we deploy four Quorum nodes on the aforementioned server and prepare transactions on the same server. We derive that fingerprints, along with associated metadata, e.g., shipment or sensor IDs, result in a payload of 124 B. Along with the transaction overhead, including headers and signatures, a single fingerprint per transaction hence requires 267 B. Preparing a transaction (TX) with a single fingerprint—including the calculation of signatures—only takes 20 ms (on average over 1000 runs) on our lightweight server. Hence, we are able to prepare 50 TX/s, which greatly exceeds our needs. In a real-world setting, this functionality would likely even run on a more powerful (per-entity) server.

Augmenting and confirming our assessment, Baliga et al. [51] have shown that submitting prepared transactions to a Quorum blockchain can very well result in a throughput of 740 TX/s, which suffices for the use cases presented in Section III-C: Each shipment only requires a single registration fingerprint along with two fingerprints (initialization, finalization) per involved sensor. Irrespective of any time constraints, merging multiple fingerprints in a single transaction, utilizing meta-fingerprints that cover multiple shipment events, sidechains, and sharding are concepts to further scale such a system performance- and storage-wise [33], [52], [53]. Specifically, prior work on supply chain information systems already has a more elaborate discussion on this matter [33].

Alternatively, the involved stakeholders can also agree on another form of tamperproof storage, e.g., relying on a single trusted entity, as we have discussed in Section IV-E.

This performance evaluation underlines the feasibility and scalability of our E2E sensing: computationally-wise, corresponding solutions are appropriate for real-world use.

B. HARDWARE DEPLOYMENT & COST ESTIMATES

Goods are typically shipped in crates, with several crates fitting into a standard container. Monitoring equipment can be installed in crates or containers, depending on the required granularity of monitoring and the trust relationships between crate owners and shipping companies. To map the use cases and requirements from Table 1, less than five sensors would be required per crate, which can, in most cases, be operated even by a single lightweight Sancus processor. Realistically, such a setup, including basic sensors, can be built for approximately 10 € per crate utilizing off-the-shelf IoT sensors. Slightly more expensive sensors may be necessary if more demanding use cases such as visual monitoring are required.

Concerning computational hardware, Sancus processors are not commercially available. However, Sancus is based on the MSP430 family of processors which can give a reasonable cost estimate if this processor would be deployed. Specifically, the Texas Instruments MSP430FR6920 family of processors allows for a reasonable comparison as it provides similar features with a cryptographic unit and memory isolation capabilities that are close in their nature to a TEE [54]. According to the authoritative Texas Instruments listing, these processors are available for under 2 € [55]. The more powerful, TrustZone-enabled ARM microcontrollers, are offered commercially for under 100 €. Container and crate equipment would rely on extended storage and processing capabilities in a data center or in the cloud [56]. Cloud-deployed TEE infrastructure, which could serve as the computational bridge, is commercially available at marginally higher prices than today's commonly-used cloud infrastructures [57].

Importantly, existing equipment from today's supply chain monitoring systems, specifically sensors and often also processing components, may be reused following our approach. That is, the use of TEEs puts no specific requirements on individual sensors, while many more recently purchased microcontrollers support TrustZone functionality [26]. Techniques to seamlessly integrate and reuse sensors in such scenarios are commercially available [58], [59], and research to provide strong security on low-end processors that is orthogonal to TEEs [60] might provide viable solutions to extend the lifespan of existing equipment. For TEE-based equipment, re-deployment and re-attestation of sensing and processing software provide strong guarantees of system integrity even after a potential runtime compromise, thereby further extending the lifespan of a deployed setup. Specifically, regarding equipment costs and operational expenses, our approach thus satisfies G4 (affordability) and keeps costs minimal.

Importantly, deployments need to account for extended shipment periods of several weeks to months, potentially without external connectivity and also without external power supply. For these periods, optional gateway equipment (cf. Section IV-C) can provide the necessary data storage, up to several GiB for each connected sensor. Batteries in a crate or container also require projections and planning of worst-case scenarios, e.g., two to three months for door-to-door shipping between China and Europe. Ultra-low power equipment, such as Sancus-based sensors, would typically consume less than 1 mA when active and only a few µA in sleep mode.[1] Thus, these lightweight sensors can potentially operate for several months on a conventional 2 Ah AA battery cell.

ARM-based sensors and gateway equipment, in particular devices that provide permanent wireless connectivity, might consume several hundred mA and necessitate battery capacities of around 1000 Ah to operate without interruption over a period of two months. Specialized equipment based on TrustZone-enabled low-power processors, such as the ARM Cortex-M23 and specialized low-power WiFi, might reduce this need substantially. Our approach introduces very limited additional processing and communication overheads in comparison with less trustworthy monitoring solutions. Therefore, our proposal for TEE-based end-to-end security in supply chains will only marginally increase the power consumption of remote sensing equipment in shipments.

VI. ASSESSING THE CONCEPT'S IMPACT

While Section V confirms the general technical applicability and financial feasibility of our design, we now discuss its capabilities and limitations. Particularly, we revisit the design goals (Section III-E) and assess whether and how our design achieves them regarding the varying requirements of different use cases (cf. Section III-C).

A. TRUST & SECURITY DISCUSSION

In Section V, we have already discussed G4 (affordability) and G5 (latency). Even in S3, the most complex scenario, appropriate latencies and data throughput is achievable while maintaining acceptable costs. Thus, we now focus on the achieved security guarantees, i.e., dedicated attacks by malicious parties, regardless of their involvement in the sensing and data processing. Instead of technical measures against fundamental attack vectors, we can also require a respective trust relationship between the involved parties (cf. Table 2).

1) MEASUREMENT MANIPULATION

In our context, data tampering corresponds to the manipulation of sensor measurements, e.g., to cover for issues during a shipment. The utilization of trusted sensors as of S1 with support for remote attestation enables unequivocal detection of such software manipulations. Moreover, manipulation of sensor measurements is always detectable in accordance with G1 (tamperproofness) and, at the same time, achieves G2 (authenticity).

2) MEASUREMENT WITHHOLDING

Data hiding can be attempted if direct data tampering is not possible, either by withholding or deleting sensor measurements. Since each sensor numbers its measurements, any withholding is generally apparent as a gap in numberings. Our design further introduces online phases (as of S2) to ensure that (trusted) sensors report the total number of conducted measurements, allowing each party to reliably verify measurement completeness (G3).

3) RETROACTIVE DATA REMOVAL AND MANIPULATION

As another threat, we also need to consider and discuss the hiding of complete sensors. In supply chain scenarios where parties cannot trust the database operator, our design S3 prevents retroactive data removal by recording fingerprints of essential sensor information on the tamperproof storage. These fingerprints serve as proof of existence and integrity protection since every deletion or manipulation of a data record from the database is noticeable, thus providing G1 (tamperproofness) and G3 (completeness).

4) DATA FORGING AND REPLAYING

Finally, malicious parties could attempt to insert forged information, i.e., submit measurements that originate from unauthentic and unrelated sensors or are made up entirely (data injection). First, as our design involves a PKI for sensors, the origin and authenticity (G2) of submitted data are ensured, while unique and attested measurement IDs per sensor prevent replay attacks, as duplicated IDs would be detectable. Hence, retroactive inclusion of information would require the deletion of sensor registrations for shipments in combination with authenticating the forged data. Second, while the ledger prevents manipulations of existing data

[1] Data on power consumption stems from commercial TI-MSP430 products [61], [62]. Sancus [28] is based on the openMSP430 core [63] but not a commercially available processor architecture, which, running on FPGA hardware for prototyping purposes, currently has incomparable power consumption characteristics.

records (G1 and G3), the trusted sensors ensure that all signatures originating from the sensor are solely for data originating from that sensor and not from the outside, preventing corresponding attacks.

Our design fulfills the outlined design goals (Section III-E) while providing a tunable trade-off between data reliability features (G1-G3), deployment and operational costs (G4), and general performance (G5) based on existing (trust) relationships and use case-specific data reliability requirements.

B. LIMITATIONS WITH END-TO-END SENSING

While E2E sensing shows potential for large-scale adoption, certain limitations still need to be overcome. Moreover, organizations must deal with issues inherent to the mirroring of physical events in digital management systems. In the following, we specifically address physical attacks on deployed sensors, as well as malicious shipment providers that either cheat on their mapping of sensors to shipment or cheat by registering duplicate sets of sensors to hide incidents.

1) DIRECT PHYSICAL ATTACKS

Employing trusted hardware is a useful countermeasure against compromised software, where adversaries can perform software-based attacks on co-located programs on the same device. This adversary model is sufficient if the device is residing in a physically protected environment where access to the hardware can either be controlled or at least be monitored. However, trusted sensors in a supply chain are naturally deployed in untrusted environments. This situation requires expanding the adversary model into the physical sphere, i.e., to account for attackers gaining physical access to the devices. For secure hardware, specifically for TEEs, physical access has been proven to grant attackers powerful privileges, such as performing glitching attacks and bus snooping [64].

On Intel SGX, a well-researched, commercial TEE, researchers have proven that voltage glitching attacks are possible and have successfully performed fault-injection attacks to retrieve cryptographic keys from enclaves [64]. For Intel SGX, such physical attacks are out of the scope of the threat model [64] and deploying a server in a data center requires safeguarding the device against physical tampering. Similarly, physical attacks are usually also out of scope when deploying embedded devices, e.g., for Sancus [28] and TrustZone [26]. This exclusion from the adversary model may be necessary and realistic for devices deployed in a controlled environment but fails to address all the necessary nuances that arise in our heterogeneous scenarios. For example, if an embedded sensing device can be physically attacked to leak its cryptographic key material, the adversary could fully impersonate it and its secure hardware element. Such attacks would invalidate guarantees that the TEE should provide and allow for all of the misuse cases described in Section III-D.

To consider all threat vectors, sensing parties should account for such attacks, e.g., by using tamperproof physical isolation of the digital components or by physically hardening their equipment [65], [66]. Most countermeasures will either require making the device tamper-evident, i.e., to enable the verification of the physical integrity of all devices for involved stakeholders, or triggering the device self-destruct when physical tampering is detected. We see these mitigations as possible but non-trivial future work, i.e., to augment our work and thereby extend the available security guarantees.

2) SENSOR REGISTRATION FORGERY

Trust into any reported sensor data can only be achieved if the underlying sensing devices are trustworthy. If an adversary can implant their own devices into the shipment from the beginning of the deployment process, then the existence of the whole chain of trust is in question. As a first step to mitigating sensor-related mistrust, we suggest only utilizing devices that have undergone some process to verify their legitimacy. This process would ideally include some general device certification, but in the absence of such, any verification process that is trusted by all stakeholders suffices. While this mitigation takes the first steps to ensure that no tampered or forged devices can be deployed by any stakeholder, some remaining issues persist: Most prominently, malicious shipment providers can register multiple sensor sets per shipment and ensure that, even if the first sensor set records issues with the shipment, the backup sensor set remains in a healthy (shielded) environment. Before handing off the shipment, a malicious shipment provider could then remove the faulty sensors to only report the shielded sensors' data. Similarly, malicious shipment providers can place sensors in a manipulated environment that differs from the intended one, i.e., conduct *physical sensor manipulation* [67]. To mitigate both issues, we suggest that future work investigates how to (semantically) verify measurements by sourcing environmental conditions of nearby sensors or different sensor types. Related work on consumer IoT already considers this research angle [68]. A similar approach could be used to also verify sensors in nearby or previous shipments.

3) PRODUCT REGISTRATION FORGERY

Malicious shipment providers could attempt to cheat by being dishonest in mapping sensors to physically shipped products [67]. In this case, the shipment provider could pretend to provide sensor data to a shipment but then only attach the sensors to the shipping when the shipment arrives at the next destination. Here, we envision different mitigation strategies. First, companies could utilize sensors that make it exceedingly difficult to replace them unnoticeably. Depending on the type of shipment, this strategy might be costly due to the need for specialized equipment. Second, the linking of physical products to digital data could be strengthened through various means [69], [70], [71]. Modern marking approaches like molecular fingerprinting [72], [73] exemplify how to uniquely identify shipments. Embedding this unique identifier into the sensor metadata then allows actors to verify the binding between sensors and shipped products. We leave the corresponding feasibility and affordability studies to future work.

While these aspects are relevant to keep in mind when deploying our proposed E2E sensing, they are also apparent in traditional sensing infrastructures, i.e., they are not specific to our work. Moreover, we argue that any real-world use would inherently require human decision-makers in the loop to make judgments and decisions based on the output of the technical domain. This addition is not only necessary to handle any misbehavior but also to accommodate potential technical failures of the E2E-secured sensing.

C. RELATED WORK: LACK OF E2E-SECURED PROCESSING

Certainly, the concept of end-to-end encryption is well-known from communication systems, such as TLS [74], and application-tailored designs in different domains, e.g., CPS [75], IoT [76], smart grids [77], among others. While these concepts focus on communication paths, their settings do not cover the data acquisition (or sensing). To the best of our knowledge, except for our prior work [17] (cf. Section I), related work has not proposed comparable approaches that also cover the secure sensing of information as part of their E2E model (i.e., communication path). Our research on securing the processing of sensed information along the full communication path, from sensor to storage, thus lies at the intersection of three topics: *supply chains*, *trusted computing*, and *blockchain technology*. Next, we discuss their influences and shortcomings concerning our proposed design.

1) SUPPLY CHAINS

Business-oriented research in the context of supply chains and their management considers a multitude of research directions [1], [2], [8]. The closest overlap with computer science concerns the processing of data. Most prominently, related work puts great emphasis on tracking and tracing in supply chains [33], [78], protecting against counterfeiting [79], [80], or recording ownership transfers [32], [81], [82], [83], with only a few approaches proposing end-to-end-secured information flows [84]. Such approaches have in common that they focus on the secure processing of information or identifiers: To date, barely any work considers the tamperproof and authentic sensing of data (G1 & G2) [17]. Simply deploying off-the-shelf IoT sensors is insufficient, especially in settings with business-oriented, distrusting stakeholders who might even have a (monetary) incentive to cheat. Thus, we require concepts to securely and reliably sense data in supply chains.

2) TRUSTED COMPUTING

Generally, a wide range of trusted execution environments, both as research projects and as commercial products, is available [24], [25]. For trusted sensors, we identify Sancus [28] and TrustZone [26] as suitable TEEs, as both can be deployed for low costs (G4) while achieving the discussed performance requirements (G5). Other embedded security architectures [85], [86] may similarly suffice as long as they provide isolation and attestation primitives. Similarly, the utilized attestation protocol and implementation are equally interchangeable, and respective related work on improving remote attestation can be integrated [15].

3) BLOCKCHAIN TECHNOLOGY

Supply chains are a prominent application area for blockchains: Related work frequently utilizes blockchain technology, most commonly when enabling or improving tracking and tracing of products or allowing for (origin) certification of products as well as counterfeit identification [32], [33], [78], [87]. While these approaches deal with the tamperproof storing and long-term availability, i.e., completeness of data, they are susceptible to "garbage-in, garbage-out" [88], as they *unreasonably assume* complete, authentic, and untampered (sensor) data as input. These shortcomings also hold for commercial products, such as the recently discontinued TradeLens [89], upkeep [12], project44 [13], or roambee [14]. Generally, they largely focus on ensuring interoperability and quality-of-operation improvements for businesses [90] while neglecting the reliable and authentic foundation of sensed data, which they indirectly build on.

Thus, we are convinced that a full-fledged end-to-end processing design is both missing and essential. Without a specific focus on supply chains, other research intersecting the digital and physical world looks into combining IoT devices and blockchain technology without deploying specifically secured devices or sensors [91], [92]. As such, their work is complementary, and our design with trusted sensors can potentially benefit from corresponding advances, primarily through affordability and latency improvements (G4 & G5).

4) CONCEPTUAL INTERSECTIONS

Several projects combine trusted computing and blockchain technology. For example, Microsoft CCF [45] realizes a replicated ledger inside a TEE. Moreover, BTAA [93] improves the cross-domain authentication in the IoT using both technologies. Other approaches explore TEEs for blockchain applications [94], [95], [96], while first projects rely on TEEs in real-world blockchain deployments [97]. Prior work [17], [67] already raised concerns about insecure sensors and unreliable data processing for supply chains and blockchains. To the best of our knowledge, a usable design utilizing specially secured sensors, i.e., trusted computing hardware, is still missing.

To conclude, we augment prior work by applying trusted sensors, TEEs, and blockchain technology for practical use in supply chains to provide technical guarantees. Thereby, we reshape today's trust boundaries in supply chains.

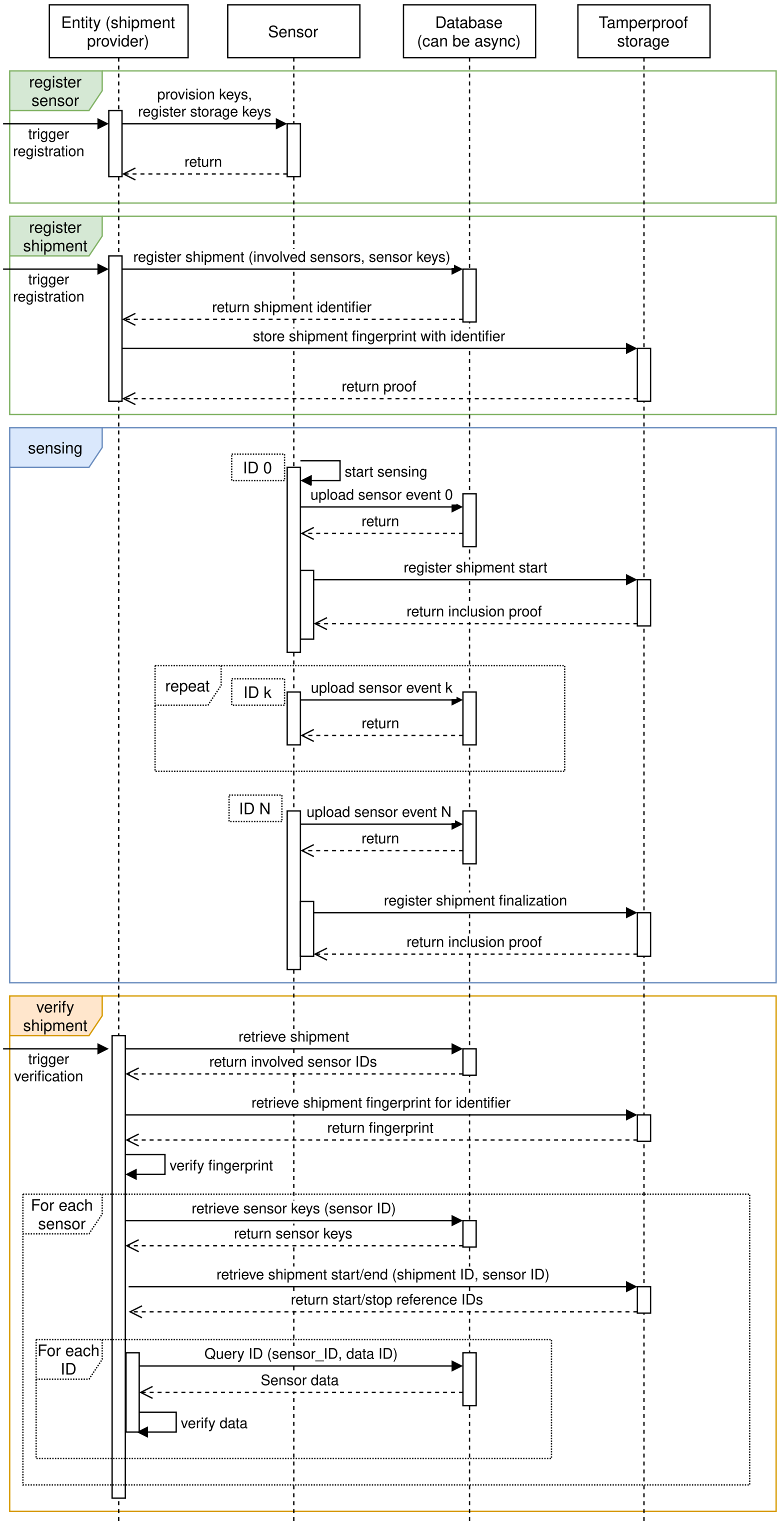

FIGURE 4. Sequence diagram for S3—Tamperproof Storage: The figure shows the complete sequence diagram of S3. After independent phases to register multiple sensors and a shipment, sensing can start after an online phase. A similar online phase is required to finish the sensing, requiring both beginning and ending proofs to verify the shipment.

D. UNIVERSALITY OF OUR SENSING CONCEPTS

Even though the primary focus of our work is on the use of E2E sensing in supply chains, and we motivated our work accordingly, our design and the corresponding discussions are also relevant to other application areas. In particular, we can easily translate the foundation of our work to settings where mutually distrustful parties sense information in (remote) environments. More specifically, suitable areas are the application of shared inventory management, rental or parking services, digitized construction sites, and smart manufacturing. Especially the latter demands accurate processing of usage and state information due to the emergence of digital factories, where manufacturing equipment and raw material are shared between companies. Depending on the exact industry, accurate monitoring of tool wear is crucial to avoid significant damages (and, in turn, costs) to the production line. Consequently, we argue that our work and the presented findings have the potential to also impact applications that exceed the "simple" tracking and monitoring in supply chains.

VII. CONCLUSION

The growing complexity in supply chains comes with a need for extensive monitoring of goods and shipments and an increase in involved stakeholders, not all of whom trust each other. Hence, for such distributed settings, approaches for secure and reliable embedded computing infrastructures, specifically regarding communications and sensing, are needed. To alleviate this situation, we presented and discussed four designs that expand on the existing situation of trust in monitoring equipment and data, and that increasingly take adversarial stakeholders into account. By utilizing trusted hardware at the sensor side and securely communicating the sensed and attested data into a database, which is backed by a tamperproof storage, our design achieves a high level of data integrity at minimal costs with real-world applicability. Our final design is, to a large extent, capable of reusing sensing equipment in shipment and supply-chain monitoring and builds on trusted execution primitives to establish and maintain trust over such equipment for an extended life span. Despite our focus on supply-chain management, our research and the resulting comparison of proposals for secure system designs can inform the development of dependable distributed systems with many stakeholders, across other domains in the context of IoT and CPS. E.g., in smart factories or in the context of the Internet of Production.

Overall, we show that realizing E2E-secured sensing is feasible and move the trust in processed data to the edge of the sensing while establishing trustworthy long-term availability of sensed data. These guarantees both apply to scenarios where shipments maintain mobile connectivity and transmit data continuously as well as to shipments that remain offline for large periods of the transit process, as is common in international freight handling. We provide an implementation and a comprehensive evaluation of the E2E-secured scenario, highlighting the feasibility, performance, and scalability aspects of all essential components. Our work fills a gap in literature and contributes a detailed discussion of design choices for trustworthy and robust supply-chain sensing that utilizes authenticated communication, a performance evaluation of crucial components necessary for real-world deployments, and a discussion of relevant pitfalls and challenges. The availability of E2E-secured sensing could also impact the reliability of reputation systems in supply chains [98], resulting in a higher degree of automation and more trust among stakeholders. We look forward to seeing respective advances in both academia and industry. Finally, our evaluation artifacts [23] are open source to support additional research in the area.

APPENDIX

SEQUENCE DIAGRAM OF S3

In Figure 4, we illustrate the complete sequence diagram of S3. After independent phases to register multiple sensors and a shipment, sensing can start after an online phase. A similar online phase is required to finish the sensing, requiring both beginning and ending proofs to verify the shipment.

ACKNOWLEDGMENT

Eduard Vlad supported the artifact creation of the blockchain performance evaluation. The authors further thank the reviewers, Bert Lagaisse, Maria Spiß, and Johannes Lohmöller for their insightful comments that allowed them to improve their work.

REFERENCES

[1] H. Fatorachian and H. Kazemi, "Impact of Industry 4.0 on supply chain performance," *Prod. Planning Control*, vol. 32, no. 1, pp. 63–81, Jan. 2020.

[2] A. Sheel and V. Nath, "Effect of blockchain technology adoption on supply chain adaptability, agility, alignment and performance," *Manage. Res. Rev.*, vol. 42, no. 12, pp. 1353–1374, Dec. 2019.

[3] J. Pennekamp, M. Henze, S. Schmidt, P. Niemietz, M. Fey, D. Trauth, T. Bergs, C. Brecher, and K. Wehrle, "Dataflow challenges in an Internet of Production: A security & privacy perspective," in *Proc. ACM Workshop Cyber-Physical Syst. Secur. Privacy (CPS-SPC)*, 2019, pp. 27–38.

[4] T. M. Mofokeng and R. Chinomona, "Supply chain partnership, supply chain collaboration and supply chain integration as the antecedents of supply chain performance," *South Afr. J. Bus. Manage.*, vol. 50, no. 1, pp. 1–10, Feb. 2019.

[5] J. Pennekamp, R. Glebke, M. Henze, T. Meisen, C. Quix, R. Hai, L. Gleim, P. Niemietz, M. Rudack, S. Knape, A. Epple, D. Trauth, U. Vroomen, T. Bergs, C. Brecher, A. Bührig-Polaczek, M. Jarke, and K. Wehrle, "Towards an infrastructure enabling the Internet of Production," in *Proc. IEEE Int. Conf. Ind. Cyber Phys. Syst. (ICPS)*, May 2019, pp. 31–37.

[6] H. Tran-Dang, N. Krommenacker, P. Charpentier, and D.-S. Kim, "Toward the Internet of Things for physical Internet: Perspectives and challenges," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 4711–4736, Jun. 2020.

[7] L. Pfahl and C. Moxham, "Achieving sustained competitive advantage by integrating ECR, RFID and visibility in retail supply chains: A conceptual framework," *Prod. Planning Control*, vol. 25, no. 7, pp. 548–571, May 2014.

[8] A. Chaudhuri, I. Dukovska-Popovska, N. Subramanian, H. K. Chan, and R. Bai, "Decision-making in cold chain logistics using data analytics: A literature review," *Int. J. Logistics Manage.*, vol. 29, no. 3, pp. 839–861, Aug. 2018.

[9] R. K. Singh, P. Kumar, and M. Chand, "Evaluation of supply chain coordination index in context to industry 4.0 environment," *Benchmarking, Int. J.*, vol. 28, no. 5, pp. 1622–1637, May 2021.

[10] P. Brauner, M. Dalibor, M. Jarke, I. Kunze, I. Koren, G. Lakemeyer, M. Liebenberg, J. Michael, J. Pennekamp, C. Quix, B. Rumpe, W. van der Aalst, K. Wehrle, A. Wortmann, and M. Ziefle, "A computer science perspective on digital transformation in production," *ACM Trans. Internet Things*, vol. 3, no. 2, pp. 1–32, 2022.

[11] L. Marques, "Sustainable supply network management: A systematic literature review from a knowledge perspective," *Int. J. Productiv. Perform. Manage.*, vol. 68, no. 6, pp. 1164–1190, Jul. 2019.

[12] UpKeep. (2014). *CMMS, EAM & IIoT Software by UpKeep Asset Operations Management*. [Online]. Available: https://www.upkeep.com/

[13] Project44. (2014). *Advanced Supply Chain Visibility*. [Online]. Available: https://www.project44.com/

[14] Roambee Corporation. (2014). *Better Real-Time Supply Chain Visibility & Intelligence*. [Online]. Available: https://www.roambee.com/

[15] C. Shepherd, R. N. Akram, and K. Markantonakis, "Establishing mutually trusted channels for remote sensing devices with trusted execution environments," in *Proc. 12th Int. Conf. Availability, Rel. Secur. (ARES)*. New York, NY, USA: ACM, Aug. 2017.

[16] M. Lezoche, J. E. Hernandez, M. D. M. E. Alemany Díaz, H. Panetto, and J. Kacprzyk, "Agri-food 4.0: A survey of the supply chains and technologies for the future agriculture," *Comput. Ind.*, vol. 117, May 2020, Art. no. 103187.

[17] J. Pennekamp, F. Alder, R. Matzutt, J. T. Mühlberg, F. Piessens, and K. Wehrle, "Secure end-to-end sensing in supply chains," in *Proc. IEEE Conf. Commun. Netw. Secur. (CNS)*, Jun. 2020, pp. 1–6.

[18] R. Roman, C. Alcaraz, J. Lopez, and K. Sakurai, "Current perspectives on securing critical infrastructures' supply chains," *IEEE Secur. Privacy*, vol. 21, no. 4, pp. 29–38, Jul./Aug. 2023, doi: 10.1109/MSEC.2023.3247946.

[19] J. Pennekamp, R. Matzutt, C. Klinkmüller, L. Bader, M. Serror, E. Wagner, S. Malik, M. Spiß, J. Rahn, T. Gürpinar, E. Vlad, S. J. J. Leemans, S. S. Kanhere, V. Stich, and K. Wehrle, "An interdisciplinary survey on information flows in supply chains," *ACM Comput. Surv.*, vol. 56, no. 2, pp. 1–38, Feb. 2024.

[20] R. C. Merkle, "A digital signature based on a conventional encryption function," in *Proc. 7th Conf. Theory Appl. Cryptograph. Techn. (CRYPTO)*, vol. 293. Berlin, Germany: Springer, 1987, pp. 369–378.

[21] R. Maull, P. Godsiff, C. Mulligan, A. Brown, and B. Kewell, "Distributed ledger technology: Applications and implications," *Strategic Change*, vol. 26, no. 5, pp. 481–489, Sep. 2017.

[22] *EPC Information Services (EPCIS) Standard*, Standard GS1 AISB, Release 1.2, 2016.

[23] J. Pennekamp, F. Alder, L. Bader, G. Scopelliti, K. Wehrle, and J. T. Mühlberg. (2024). *Securing Sensing in Supply Chains: Opportunities, Building Blocks, and Designs*. [Online]. Available: https://github.com/COMSYS/secure-sensing

[24] P. Maene, J. Götzfried, R. de Clercq, T. Müller, F. Freiling, and I. Verbauwhede, "Hardware-based trusted computing architectures for isolation and attestation," *IEEE Trans. Comput.*, vol. 67, no. 3, pp. 361–374, Mar. 2018.

[25] M. Schneider, R. J. Masti, S. Shinde, S. Capkun, and R. Perez, "SoK: Hardware-supported trusted execution environments," 2022, *arXiv:2205.12742*.

[26] S. Pinto and N. Santos, "Demystifying arm TrustZone: A comprehensive survey," *ACM Comput. Surv.*, vol. 51, no. 6, pp. 1–36, Nov. 2019.

[27] V. Costan and S. Devadas, "Intel SGX explained," Cryptol. ePrint Arch., Tech. Rep. 2016/086, 2016. [Online]. Available: https://eprint.iacr.org/2016/086

[28] J. Noorman, J. V. Bulck, J. T. Mühlberg, F. Piessens, P. Maene, B. Preneel, I. Verbauwhede, J. Götzfried, T. Müller, and F. Freiling, "Sancus 2.0: A low-cost security architecture for IoT devices," *ACM Trans. Privacy Secur.*, vol. 20, no. 3, pp. 1–33, Aug. 2017.

[29] G. Scopelliti, S. Pouyanrad, J. Noorman, F. Alder, C. Baumann, F. Piessens, and J. T. Mühlberg, "End-to-end security for distributed event-driven enclave applications on heterogeneous TEEs," *ACM Trans. Privacy Secur.*, vol. 26, no. 3, pp. 1–46, Aug. 2023.

[30] VERAISON. (2021). *Project VERAISON: VERificAtIon of AtteStatiON*. [Online]. Available: https://github.com/veraison

[31] A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*. Princeton, NJ, USA: Princeton Univ. Press, 2016.

[32] S. Malik, S. S. Kanhere, and R. Jurdak, "ProductChain: Scalable blockchain framework to support provenance in supply chains," in *Proc. IEEE 17th Int. Symp. Netw. Comput. Appl. (NCA)*, Nov. 2018, pp. 1–10.

[33] L. Bader, J. Pennekamp, R. Matzutt, D. Hedderich, M. Kowalski, V. Lücken, and K. Wehrle, "Blockchain-based privacy preservation for supply chains supporting lightweight multi-hop information accountability," *Inf. Process. Manage.*, vol. 58, no. 3, May 2021, Art. no. 102529.

[34] F. Tian, "An agri-food supply chain traceability system for China based on RFID & blockchain technology," in *Proc. 13th Int. Conf. Service Syst. Service Manage. (ICSSSM)*, Jun. 2016, pp. 1–6.

[35] Z. Zheng, S. Xie, H.-N. Dai, X. Chen, and H. Wang, "Blockchain challenges and opportunities: A survey," *Int. J. Web Grid Services*, vol. 14, no. 4, pp. 352–375, 2018.

[36] R. Matzutt, B. Kalde, J. Pennekamp, A. Drichel, M. Henze, and K. Wehrle, "CoinPrune: Shrinking Bitcoin's blockchain retrospectively," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 3, pp. 3064–3078, Sep. 2021.

[37] L. S. Sankar, M. Sindhu, and M. Sethumadhavan, "Survey of consensus protocols on blockchain applications," in *Proc. 4th Int. Conf. Adv. Comput. Commun. Syst. (ICACCS)*, Jan. 2017, pp. 1–5.

[38] J. Byun, S. Woo, and D. Kim, "Efficient and privacy-enhanced object traceability based on unified and linked EPCIS events," *Comput. Ind.*, vol. 89, pp. 35–49, Aug. 2017.

[39] *EPCIS Standard*, Standard GS1 AISBL, Release 2.0, Community Review Draft, 2021.

[40] *Standardization of Data and Documentation Practices for Product Tracing Guidance for Industry*, Food Drug Admin., White Oak, MD, USA, document FDA-2018-D-0688, 2018.

[41] D.-L. Wu, W. W. Y. Ng, D. S. Yeung, and H.-L. Ding, "A brief survey on current RFID applications," in *Proc. Int. Conf. Mach. Learn. Cybern.*, vol. 4, Jul. 2009, pp. 2330–2335.

[42] D. Dolev and A. C. Yao, "On the security of public key protocols," *IEEE Trans. Inf. Theory*, vol. IT-29, no. 2, pp. 198–208, Mar. 1983.

[43] M. D. Ryan, "Enhanced certificate transparency and end-to-end encrypted mail," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, 2014, pp. 1–14. [Online]. Available: see https://www.ndss-symposium.org/wp-content/uploads/2017/09/12_2_1.pdf

[44] M. Paik, J. Irazábal, D. Zimmer, M. Meloni, and V. Padurean, "immudb: A lightweight, performant immutable database," CodeNotary, Bellaire, TX, USA, Tech. Rep., 2020. [Online]. Available: https://codenotary.s3.amazonaws.com/Research-Paper-immudb-CodeNotary_v3.0.pdf

[45] M. Russinovich, E. Ashton, C. Avanessians, M. Castro, A. Chamayou, S. Clebsch, M. Costa, C. Fournet, M. Kerner, S. Krishna, J. Maffre, T. Moscibroda, K. Nayak, O. Ohrimenko, F. Schuster, R. Schuster, A. Shamis, O. Vrousgou, and C. M. Wintersteiger, "CCF: A framework for building confidential verifiable replicated services," Microsoft, Redmond, WA, USA, Tech. Rep. MSR-TR-2019-16, 2019.

[46] XESS Corporation. (2015). *StickIt! Main Motherboard*. [Online]. Available: https://github.com/xesscorp/StickIt-MB

[47] Sancus. (2022). *Sancus-Core*. [Online]. Available: https://github.com/sancus-tee/sancus-core/releases/tag/v2.1.0

[48] S. Rautmare and D. M. Bhalerao, "MySQL and NoSQL database comparison for IoT application," in *Proc. IEEE Int. Conf. Adv. Comput. Appl. (ICACA)*, Oct. 2016, pp. 235–238.

[49] CONSENSYS. (2020). *ConsenSys Quorum*. [Online]. Available: https://consensys.net/quorum/

[50] S. De Angelis, L. Aniello, F. Lombardi, A. Margheri, and V. Sassone, "PBFT vs proof-of-authority: Applying the CAP theorem to permissioned blockchain," in *Proc. 2nd Italian Conf. Cyber Secur. (ITASEC)*, vol. 2058, 2017, pp. 1–11. [Online]. Available: https://ceur-ws.org/Vol-2058/paper-06.pdf

[51] A. Baliga, I. Subhod, P. Kamat, and S. Chatterjee, "Performance evaluation of the quorum blockchain platform," 2018, *arXiv:1809.03421*.

[52] M. Zamani, M. Movahedi, and M. Raykova, "RapidChain: Scaling blockchain via full sharding," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2018, pp. 931–948.

[53] A. Back, M. Corallo, L. Dashjr, M. Friedenbach, G. Maxwell, A. Miller, A. Poelstra, J. Timón, and P. Wuille, "Enabling blockchain innovations with pegged sidechains," Blockstream, Victoria, BC, Canada, Tech. Rep., 2014. [Online]. Available: https://blockstream.com/sidechains.pdf

[54] Texas Instruments. (2018). *MSP430FR6920 Datasheet*. Accessed Dec. 18, 2023. [Online]. Available: https://www.ti.com/product/MSP430FR6920

[55] Texas Instruments. (2023). *MSP430 Microcontrollers*. Accessed: Dec. 18, 2023. [Online]. Available: https://www.ti.com/microcontrollers-mcus-processors/msp430-microcontrollers/products.html

[56] M. Henze, "The quest for secure and privacy-preserving cloud-based industrial cooperation," in *Proc. IEEE Conf. Commun. Netw. Secur. (CNS)*, Jun. 2020, pp. 1–5.

[57] Intel Corporation. (2022). *Intel SGX Product Offerings*. Accessed: Dec. 18, 2023. [Online]. Available: https://www.intel.com/content/www/us/en/architecture-and-technology/sgx-product-offerings.html

[58] F. Yang, N. Matthys, R. Bachiller, S. Michiels, W. Joosen, and D. Hughes, "*mu*PnP: Plug and play peripherals for the Internet of Things," in *Proc. 10th Eur. Conf. Comput. Syst. (EuroSys)*, Apr. 2015, pp. 1–14, Art. no. 25, doi: 10.1145/2741948.2741980.

[59] E. Aras, S. Delbruel, F. Yang, W. Joosen, and D. Hughes, "A low-power hardware platform for smart environment as a call for more flexibility and re-usability," in *Proc. Int. Conf. Embedded Wireless Syst. Netw. (EWSN)*, 2019, pp. 194–205.

[60] J. Maerien, S. Michiels, D. Hughes, C. Huygens, and W. Joosen, "SecLooCI: A comprehensive security middleware architecture for shared wireless sensor networks," *Ad Hoc Netw.*, vol. 25, pp. 141–169, Feb. 2015.

[61] Z. Albus, A. Valenzuela, and M. Buccini, "Ultra-low power comparison: MSP430F2x MCUs vs. microchip XLP tech brief," Texas Instruments, Dallas, TX, USA, White Paper SLAY015A, 2009.

[62] *MSP430 Ultra-Low-Power Microcontrollers*, document SLAB034A, Texas Instruments, Dallas, TX, USA, 1999.

[63] O. Girard. (2009). *openMSP430—A Synthesizable 16 Bit Microcontroller Core Written in Verilog*. [Online]. Available: https://opencores.org/projects/openmsp430

[64] Z. Chen, G. Vasilakis, K. Murdock, E. Dean, D. Oswald, and F. D. Garcia, "VoltPillager: Hardware-based fault injection attacks against Intel SGX enclaves using the SVID voltage scaling interface," in *Proc. 30th USENIX Secur. Symp. (SEC)*. Berkeley, CA, USA: USENIX Association, 2021, pp. 699–716.

[65] R. Anderson and M. Kuhn, "Low cost attacks on tamper resistant devices," in *Proc. 5th Int. Workshop Secur. Protocols (Security Protocols)*, vol. 1361. Berlin, Germany: Springer, 1997, pp. 125–136.

[66] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Proc. 19th Annu. Int. Cryptol. Conf. (CRYPTO)*, vol. 1666. Berlin, Germany: Springer, 1999, pp. 388–397.

[67] K. Wüst and A. Gervais, "Do you need a blockchain?" in *Proc. Crypto Valley Conf. Blockchain Technol. (CVCBT)*, Jun. 2018, pp. 45–54.

[68] S. Birnbach, S. Eberz, and I. Martinovic, "Haunted house: Physical smart home event verification in the presence of compromised sensors," *ACM Trans. Internet Things*, vol. 3, no. 3, pp. 1–28, Aug. 2022.

[69] S. Pollard, G. Adams, F. Azhar, and F. Dickin, "Authentication of 3D printed parts using 3D physical signatures," in *Proc. NIP & Digit. Fabr. Conf., Printing Fabr.* Springfield, VA, USA: Society for Imaging Science and Technology, 2018, pp. 196–201.

[70] D. M. S. Velandia, N. Kaur, W. G. Whittow, P. P. Conway, and A. A. West, "Towards industrial Internet of Things: Crankshaft monitoring, traceability and tracking using RFID," *Robot. Comput.-Integr. Manuf.*, vol. 41, pp. 66–77, 2016.

[71] IBM Research. (2019). *Changing the Way the World Works: IBM Research's '5 in 5'*. Accessed: Dec. 18, 2023. [Online]. Available: https://research.ibm.com/blog/ibm-research-5-in-5-2018

[72] A. P. Sobolev, S. Circi, D. Capitani, C. Ingallina, and L. Mannina, "Molecular fingerprinting of food authenticity," *Current Opinion Food Sci.*, vol. 16, pp. 59–66, Aug. 2017.

[73] A. Capecchi, D. Probst, and J.-L. Reymond, "One molecular fingerprint to rule them all: Drugs, biomolecules, and the metabolome," *J. Cheminformatics*, vol. 12, no. 1, p. 43, Dec. 2020.

[74] X. de Carné de Carnavalet and P. C. van Oorschot, "A survey and analysis of TLS interception mechanisms and motivations: Exploring how end-to-end TLS is made 'end-to-me' for Web traffic," *ACM Comput. Surv.*, vol. 55, no. 13s, pp. 1–40, Dec. 2023.

[75] M. Dahlmanns, J. Pennekamp, I. B. Fink, B. Schoolmann, K. Wehrle, and M. Henze, "Transparent end-to-end security for publish/subscribe communication in cyber-physical systems," in *Proc. ACM Workshop Secure Trustworthy Cyber-Phys. Syst.*, Apr. 2021, pp. 78–87.

[76] P. Porambage, A. Braeken, A. Gurtov, M. Ylianttila, and S. Spinsante, "Secure end-to-end communication for constrained devices in IoT-enabled ambient assisted living systems," in *Proc. IEEE 2nd World Forum Internet Things (WF-IoT)*, Dec. 2015, pp. 711–714.

[77] T. Sauter and M. Lobashov, "End-to-end communication architecture for smart grids," *IEEE Trans. Ind. Electron.*, vol. 58, no. 4, pp. 1218–1228, Apr. 2011.

[78] P. Gonczol, P. Katsikouli, L. Herskind, and N. Dragoni, "Blockchain implementations and use cases for supply chains—A survey," *IEEE Access*, vol. 8, pp. 11856–11871, 2020.

[79] K. Toyoda, P. T. Mathiopoulos, I. Sasase, and T. Ohtsuki, "A novel blockchain-based product ownership management system (POMS) for anti-counterfeits in the post supply chain," *IEEE Access*, vol. 5, pp. 17465–17477, 2017.

[80] N. Anita, M. Vijayalakshmi, and S. M. Shalinie, "Blockchain-based anonymous anti-counterfeit supply chain framework," *Sādhanā*, vol. 47, no. 4, p. 208, Oct. 2022.

[81] J. Pennekamp, L. Bader, R. Matzutt, P. Niemietz, D. Trauth, M. Henze, T. Bergs, and K. Wehrle, "Private multi-hop accountability for supply chains," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Jun. 2020, pp. 1–7.

[82] S. Malik, N. Gupta, V. Dedeoglu, S. S. Kanhere, and R. Jurdak, "TradeChain: Decoupling traceability and identity in blockchain enabled supply chains," in *Proc. IEEE 20th Int. Conf. Trust, Secur. Privacy Comput. Commun. (TrustCom)*, Oct. 2021, pp. 1141–1152.

[83] M. Vijayalakshmi, S. M. Shalinie, M. H. Yang, S.-C. Lai, and J.-N. Luo, "A blockchain-based secure radio frequency identification ownership transfer protocol," *Secur. Commun. Netw.*, vol. 2022, pp. 1–12, Jan. 2022.

[84] A. Mosteiro-Sanchez, M. Barcelo, J. Astorga, and A. Urbieta, "End to end secure data exchange in value chains with dynamic policy updates," 2022, *arXiv:2201.06335*.

[85] P. Koeberl, S. Schulz, A.-R. Sadeghi, and V. Varadharajan, "TrustLite: A security architecture for tiny embedded devices," in *Proc. 9th Eur. Conf. Comput. Syst. (EuroSys)*, 2014, pp. 1–14, Art. no. 10, doi: 10.1145/2592798.2592824.

[86] I. D. O. Nunes, K. Eldefrawy, N. Rattanavipanon, M. Steiner, and G. Tsudik, "VRASED: A verified hardware/software co-design for remote attestation," in *Proc. 28th USENIX Secur. Symp. (SEC)*. Berkeley, CA, USA: USENIX Association, 2019, pp. 1429–1446.

[87] S. Malik, V. Dedeoglu, S. S. Kanhere, and R. Jurdak, "TrustChain: Trust management in blockchain and IoT supported supply chains," in *Proc. IEEE Int. Conf. Blockchain (Blockchain)*, Jul. 2019, pp. 184–193.

[88] W. Powell, M. Foth, S. Cao, and V. Natanelov, "Garbage in garbage out: The precarious link between IoT and blockchain in food supply chains," *J. Ind. Inf. Integr.*, vol. 25, Jan. 2022, Art. no. 100261.

[89] TradeLens. (2018). *TradeLens*. [Online]. Available: https://www.tradelens.com/

[90] T. Jensen, J. Hedman, and S. Henningsson, "How TradeLens delivers business value with blockchain technology," *MIS Quart. Executive*, vol. 18, no. 4, pp. 221–243, Dec. 2019.

[91] V. Dedeoglu, R. Jurdak, G. D. Putra, A. Dorri, and S. S. Kanhere, "A trust architecture for blockchain in IoT," in *Proc. 16th EAI Int. Conf. Mobile Ubiquitous Syst., Comput., Netw. Services (MobiQuitous)*. New York, NY, USA: ACM, 2019, pp. 190–199.

[92] M. Pincheira and M. Vecchio, "Towards trusted data on decentralized IoT applications: Integrating blockchain in constrained devices," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Jun. 2020, pp. 1–6.

[93] W. Mao, P. Jiang, and L. Zhu, "BTAA: Blockchain and TEE assisted authentication for IoT systems," *IEEE Internet Things J.*, vol. 10, no. 14, pp. 12603–12615, Jul. 2023, doi: 10.1109/JIOT.2023.3252565.

[94] M. Tran, L. Luu, M. S. Kang, I. Bentov, and P. Saxena, "Obscuro: A Bitcoin mixer using trusted execution environments," in *Proc. 34th Annu. Comput. Secur. Appl. Conf. (ACSAC)*. New York, NY, USA: ACM, 2018, pp. 692–701.

[95] R. Cheng, F. Zhang, J. Kos, W. He, N. Hynes, N. Johnson, A. Juels, A. Miller, and D. Song, "Ekiden: A platform for confidentiality-preserving, trustworthy, and performant smart contracts," in *Proc. IEEE Eur. Symp. Secur. Privacy (EuroS&P)*, Jun. 2019, pp. 185–200.

[96] S. Matetic, K. Wüst, M. Schneider, K. Kostiainen, G. Karame, and S. Capkun, "BITE: Bitcoin lightweight client privacy using trusted execution," in *Proc. 28th USENIX Secur. Symp. (SEC)*. Berkeley, CA, USA: USENIX Association, 2019, pp. 783–800.

[97] MobileCoin. (2017). *MobileCoin—Safe & Easy Payments at Light-Speed*. [Online]. Available: https://mobilecoin.com/

[98] L. Bader, J. Pennekamp, E. Thevaraj, M. Spiß, S. S. Kanhere, and K. Wehrle, "Reputation systems for supply chains: The challenge of achieving privacy preservation," in *Proc. 20th EAI Int. Conf. Mobile Ubiquitous Syst., Comput., Netw. Services (MobiQuitous)*. Cham, Switzerland: Springer, 2023.

JAN PENNEKAMP (Graduate Student Member, IEEE) received the B.Sc. and M.Sc. degrees (Hons.) in computer science from RWTH Aachen University. He is currently a Researcher with the Chair of Communication and Distributed Systems (COMSYS), RWTH Aachen University. His research interests include security and privacy aspects in the Industrial Internet of Things (IIoT), privacy-enhancing technologies, the design of privacy-preserving protocols, secure computations, and their general applications.

FRITZ ALDER (Graduate Student Member, IEEE) received the B.Sc. degree in computer science from RWTH Aachen University, Germany, the M.Sc. degree in IT-security from TU Darmstadt, Germany, and the Ph.D. degree from KU Leuven, Belgium. He is currently a Postdoctoral Researcher with KU Leuven. His research interests include confidential computing and system security, specifically designing and securing hardware-based trusted execution environments (TEEs).

LENNART BADER (Associate Member, IEEE) received the B.Sc. and M.Sc. degrees in computer science from RWTH Aachen University. He is currently a Researcher with the Cyber Analysis and Defense (CA&D) Department, Fraunhofer Institute for Communication, Information Processing and Ergonomics FKIE, where he is a member of the Secure Production and Energy Networks Group. His research interests include security in industrial networks, with a specific focus on energy networks.

GIANLUCA SCOPELLITI received the B.Sc. and M.Sc. degrees in computer engineering from Politecnico di Torino, Italy, in 2018 and 2020, respectively. He is currently pursuing the Ph.D. degree with KU Leuven, Belgium. He is a Researcher with Ericsson Security Research, Sweden. His research interests include system and network security, with a specific focus on hardware-based trusted computing technologies and trusted execution environments (TEEs).

KLAUS WEHRLE (Member, IEEE) received the Diploma (equivalent to M.Sc.) and Ph.D. degrees (Hons.) from the University of Karlsruhe (now KIT). Since 2010, he has been a Full Professor and the Head of the Chair of Communication and Distributed Systems (COMSYS), RWTH Aachen University. His research interests include (but are not limited to) the engineering of networking protocols, (formal) methods for protocol engineering and network analysis, reliable communication software, and all operating system issues of networking. Before joining RWTH Aachen University, in 2006, he was a Postdoctoral Researcher with the International Computer Science Institute (ICSI), in 2002 and 2003. Furthermore, he serves as a Representative for EE and CS on the Main Evaluation Board of the Alexander-von-Humboldt Foundation. He is a member of ACM, Sigcomm, GI, VDE, GI/ITG-Fachgruppe KuVS, and ACATECH.

JAN TOBIAS MÜHLBERG received the M.Sc. degree in computer science from the Brandenburg University of Applied Sciences, Germany, and the Ph.D. degree in computer science from the University of York, U.K. He is currently a Professor of embedded systems security with Université Libre de Bruxelles—École Polytechnique, Belgium. His research interests include the privacy, safety, and security of information and communications technologies (ICTs), with a particular interest in dependable embedded systems and secure critical ICT infrastructures, and in interdisciplinary research on questions around privacy and security for vulnerable populations and the responsible and sustainable development and use of ICTs.