Secure End-to-End Sensing in Supply Chains

Jan Pennekamp*, Fritz Alder†, Roman Matzutt*, Jan Tobias Mühlberg†, Frank Piessens†, Klaus Wehrle*
**Communication and Distributed Systems*, RWTH Aachen University, Germany · {lastname}@comsys.rwth-aachen.de
†*imec-DistriNet*, KU Leuven, Belgium · {fritz.alder, jantobias.muehlberg, frank.piessens}@cs.kuleuven.be

***Abstract*—Trust along digitalized supply chains is challenged by the aspect that monitoring equipment may not be trustworthy or unreliable as respective measurements originate from potentially untrusted parties. To allow for dynamic relationships along supply chains, we propose a blockchain-backed supply chain monitoring architecture relying on trusted hardware. Our design provides a notion of secure end-to-end sensing of interactions even when originating from untrusted surroundings. Due to attested checkpointing, we can identify misinformation early on and reliably pinpoint the origin. A blockchain enables long-term verifiability for all (now trustworthy) IoT data within our system even if issues are detected only after the fact. Our feasibility study and cost analysis further show that our design is indeed deployable in and applicable to today's supply chain settings.**

***Index Terms*—supply chain; trusted computing; trusted execution; blockchain; Internet of Production; condition monitoring**

I. Introduction

Advances in the Internet of Things (IoT), especially its comprehensive sensing capabilities, have drastically reshaped business processes and thus established new application domains, e.g., smart cities [1]. This development of deploying sensing devices is mainly fueled by cost reductions, more energy-efficient hardware, and increased processing capabilities. More recently, increased deployments of IoT devices in industrial applications, e.g., to monitor manufacturing processes [2], coined the notion of an Industrial IoT (IIoT). However, IIoT applications today mainly consider intra-corporate deployments [3], i.e., the process is controlled by a single party.

Contrary to current IIoT deployments, complex production processes rely on distributed, potentially global, supply chains, spanning vast numbers of collaborators [3], [4]. This discrepancy is further aggravated by new paradigms such as the Internet of Production (IoP) [2], which deliberately seeks to break up stiff collaborations along traditional supply chains to provide manufacturers with more flexibility. While this approach promises improved product quality and a reduced amount of scrap [3], the IoP will come at the cost of breaking established trust relationships among well-known collaborators to introduce dynamic, short-lived relationships instead.

Specifically, crucial process or production data may be generated by shipments in transit or within the facilities of external collaborators, whom the end product manufacturer may not fully trust. Hence, companies require reliable, distributed, yet secure monitoring capabilities along the whole supply chain to improve their own processes. They especially want to hold unknown, but potentially valuable collaborators and suppliers accountable if discrepancies to previously reported information surface on delivery or during production. Such trustworthy monitoring itself must remain (i) cost-efficient, (ii) easily deployable, and (iii) maintainable to constitute a sustainable alternative to the current approach of relying on legally-vetted contracts with complex and time-costly maintenance.

In this work, we thus introduce a secure *end-to-end (E2E) sensing* architecture to establish trustworthy long-term verifiability even in highly dynamic supply chains, as they are envisioned in distributed next-generation manufacturing [5]. We build upon recent advances [6] in the IIoT by incorporating *trusted sensors* [7] and *Trusted Execution Environments* (TEEs) [8] to relay and process sensor readings through trusted and tamperproof dataflows, even if those readings are gathered in hostile environments. We eventually persist all these recorded supply chain interactions and conditions immutably as checkpoints on a tamperproof *distributed ledger* based on blockchain technology [9] to provide stakeholders with a complete history of attested and verifiable checkpoints while accounting for the highly federated and flexible landscape of modern supply chain collaborators. Effectively, we establish a distributed architecture for secure and reliable E2E sensing (from sensor to storage) even in highly dynamic supply chains.

Our Contributions. Our main contributions are as follows.

- **(a)** We prepare IoT-driven supply chains for dynamic, short-lived relationships by proposing a new trustworthy end-to-end (E2E) sensing that sources data from lightweight TEE-based sensors and creates a blockchain-based log that enables long-term verifiable (condition) monitoring.
- **(b)** We assess the performance of suitable hardware components in our design for different data sources in supply chains with and without continual monitoring needs.
- **(c)** We conduct a cost analysis for such settings to give additional insights into potential deployment scenarios.

This way, we enable improvements in trust and information exchange within existing supply chains as well as foster the establishment of novel and more flexible business relationships. We further expect that our design is also applicable to other areas. For example, customs handling could be improved as trusted sensors provide detailed insights into the shipment.

II. Trusted Computing Background

Trusted computing promises to give security guarantees even when the software is running on potentially malicious devices, e.g., if they are deployed in untrusted surroundings. This goal can be achieved by utilizing hardware-based TEEs that (i) *isolate* parts of software within the device, and (ii) reliably *attest* the correctness of its computations to remote parties as a trust anchor [8]. TEEs typically also provide the additional security properties of *memory protection* and *sealing* [8].

Author manuscript.

DOI: 10.1109/CNS48642.2020.9162337

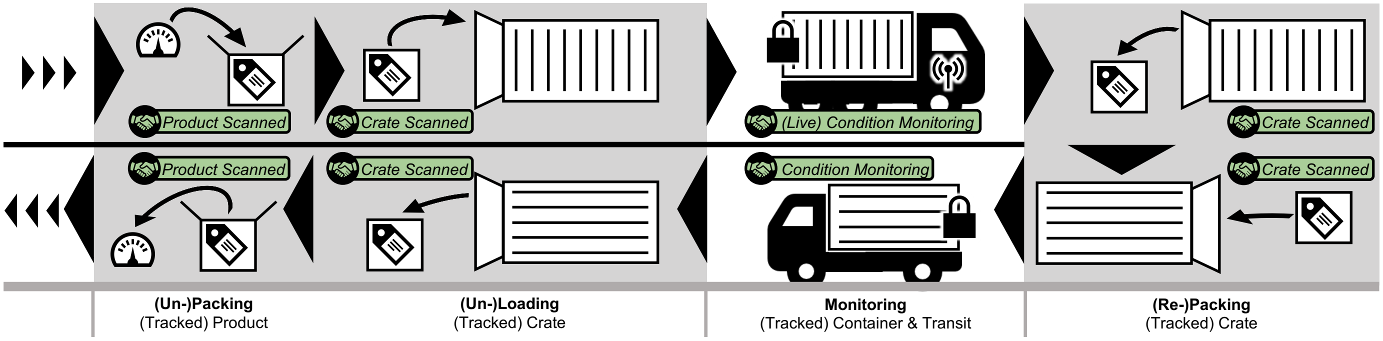

Fig. 1. In our scenario, we consider products that are (from top left to top right) packed into crates, loaded, shipped, and possibly repacked along the supply chain until they reach their recipient (bottom left). Throughout this process, the product should be tracked in a trustworthy way, i.e., an infrastructure guarantees the correctness of attested and verifiable checkpoints (green background). Additionally, its conditions can even be continually monitored securely.

TEE Implementations. In the last years, several TEE-enabled systems were designed and deployed: Intel SGX [10], which is even available on end-user processors, and ARM TrustZone [11] are two common examples. ARM TrustZone is available on higher-level IoT devices with the Cortex-A family [11] and is also widely deployed on smartphones. Sancus [12], a more lightweight TEE, bases on a low-level, low-cost processor with low resource capabilities. Therefore, it is ideally suited for wide-spread deployments on IoT nodes. While the computation capabilities of such cost-efficient TEEs may be limited, the performed computations remain secure.

A core feature of TEEs is *remote attestation*, which allows remote users to verify that the TEE is running the intended program code. In both Intel SGX and Sancus, remote attestation enables remote users to check the target platform's code against a known checksum. Furthermore, users can establish a secure channel directly into the protected module (called *enclave* for Intel SGX) if this checksum matches a trusted program version [8]. Naturally, this feature also facilitates mutual attestations, e.g., between Sancus and Intel SGX. In the following, we present TEE-backed IoT and cloud deployments in more detail as a foundation for our proposed design.

IoT-suited TEEs. Sancus is especially well-suited for the IoT as it is built on the MSP-430 [12], a 16-bit processor family. Sancus runs at a speed of 20 MHz and provides only 65 kB for code and data, which is sufficient for simple IoT applications. Furthermore, the MSP-430 supports the use of *memory-mapped I/O modules (MMIOs)*, e.g., to attach sensors as utilized in the IoT. With Sancus, these MMIO devices can be assigned to a specific protected module that guarantees isolated sensor access [7], i.e., this design enables the trusted code in the TEE to have a secure, exclusive channel to the sensor readings. As such, it allows for further processing and reporting to a remote party directly from inside the TEE.

Cloud-deployed TEEs. Due to its integration into widely available Intel CPUs, SGX is highly accessible for augmenting Internet services, where a single server can easily provide service to hundreds of IoT nodes. In contrast to the IoT-suited TEEs, SGX-created attestations are publicly verifiable by default through a dedicated attestation service, i.e., information signed by the TEE is verifiable by any party. Thus, SGX can create signatures for (trusted) data it received from remote data sources and thus make IoT-gathered data publicly verifiable.

III. Supply Chain Monitoring

In this section, we introduce our use case and scenario for secure, trusted supply chain monitoring. We sketch our overall scenario in Section III-A, and formalize the requirements introduced by our supply chain setting in Section III-B.

A. Scenario Overview

We consider a supply chain with different involved stakeholders as our general scenario. On a high level, at each processing step along the supply chain, the interactions should be persistently recorded for long-term verifiability. For example, as shown in Figure 1, products are packed into crates, which are then shipped and finally unpacked by the recipient. Such interactions are potentially relevant for involved collaborators as they might require process adjustments if issues occur.

Lack of Trust. As expected (cf. Section I), dynamic and short-lived supply chain structures might lead to a lack of trust between companies. Simply recording all interactions of transported goods is an insufficient attempt as companies want to rely on this data to integrate it into their processes. For example, any entity with access to the taken communication path can compromise recorded data. Consequentially, companies are interested in a secure and reliable solution that can transmit unaltered data records to make up for lacking trust.

Scenarios. In particular, we consider two supply chain scenarios. First, in a basic setting, companies are interested in the delivery status of an ordered product, i.e., its shipment progress. They want to know the current state of the crate or container that contains their delivery. Here, interactions must be recorded whenever the state of the product, crate, or container in question changes, e.g., whenever this crate is unloaded from a container. Second, in addition to our basic product-tracking scenario, we distinguish an extended scenario involving condition monitoring. Some products require certain environmental conditions to remain intact. Especially, upholding a cold chain for products in a food supply chain is critical. Here, the recipient requires reliable sensing data as to whether the conditions were satisfied. Hence, this scenario requires continual sensing to provide a complete condition log.

Flexible Monitoring. The variety of available low-cost sensors allows companies to tune the monitoring granularity use case-specific. For example, the humidity within a container might not be measured for each product in the shipment

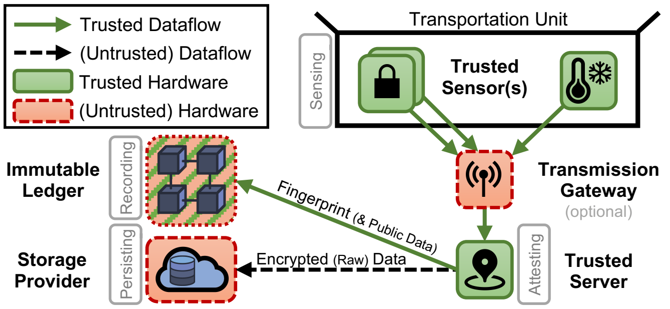

Fig. 2. As part of our architecture, we propose to deploy trusted sensors and operate a trusted server per party to establish trusted (and verifiable) dataflows from the sensors to a data storage. To this end, we record cryptographic fingerprints on the immutable ledger while the raw data is stored externally.

individually but instead once per crate. If needed, sensors can continually monitor conditions during transit, and either report the condition at specific time intervals or on-demand, e.g., by manual interaction at the next destination. Sensors can furthermore record any opening or closing of containers or crates to verify that shipments have not been tampered with during transit. We expect that the next hop, i.e., the next involved company, inspects containers and crates for drastic modifications and, thus, would report its findings accordingly.

Research Gap. Today's supply chains lack a solution that provides recipients of products with the means to rely on reported sensor data. The goal for such an approach is to improve the reliability and trustworthiness of information so that companies can rely on it within their business processes, i.e., any submitted checkpoint should be correct and untampered. To this end, collaborators require a concept for secure E2E sensing in supply chains. Such a solution would enable companies to reliably detect shipment mistakes or undesirable environmental conditions during transit even when reported from untrusted points of origin. While a missing product is detected with a delay once the shipment arrives, an interrupted cold chain might not be immediately noticeable.

In the following, we specify a set of requirements that results from our considered supply chain scenarios.

B. *Requirements for Secure E2E Supply Chain Sensing*

Based on our presented scenario, we derive four aspects that must be addressed to reliably improve today's approaches.

R1: Tamperproofness. The measured sensor data must be untampered until it arrives at the recipient so that companies can integrate this information into their business processes.

R2: Authenticity. The received sensor data must be authentic, i.e., it has to originate from the claimed sensor. Any solution must prevent the insertion of false or replayed data.

R3: Accountability. Companies should be accountable for sensed and reported information that originates from their environment, i.e., deception must be attributable. As such, all sensed data must be retained to provide verifiability.

R4: Applicability. Any proposed solution must be scalable to satisfy the sensing frequency of today's supply chains. Furthermore, potentially needed hardware should be affordable.

Based on these requirements, we now propose a novel architecture that enables secure E2E sensing in supply chains.

IV. An Architecture Introducing Trustworthy & Verifiable Supply Chain Monitoring

We now introduce our new supply chain architecture that utilizes TEEs and a blockchain to provide secure E2E sensing. First, we introduce our design overview in Section IV-A. Then, in Section IV-B, we present the different steps of our solution. Finally, we discuss deployment considerations in Section IV-C.

A. *Design Overview*

We propose to achieve secure E2E sensing by relying on trusted computing. In particular, we create a trusted dataflow from TEE-backed sensors in untrusted surroundings to a tamperproof storage, as illustrated in Figure 2. The *trusted sensors* reliably sense all interactions with or conditions of shipments along the supply chain in a verifiable and tamperproof manner, i.e., read conditions cannot be modified. These nodes attest and securely transfer their data to a *trusted server* that also operates within a TEE. There, the raw data is encrypted so that it can be securely persisted at a dedicated (cloud) *storage provider*. Finally, data intended for the public and a corresponding fingerprint of all (sensor) data, for example, a digital signature, is recorded on a decentralized, *immutable ledger*.

This design ensures a trusted dataflow from the source, originating from potentially untrusted surroundings, to a tamperproof medium where the sensed data is recorded. Hence, we can provide an end-to-end guarantee for all sensed information while providing long-term verifiability. In the following, we introduce the individual steps of our proposed supply chain architecture in more depth. Furthermore, we provide details on an optional *transmission gateway* in Section IV-C.

B. *Processing Steps within our End-to-End Sensing*

A trusted dataflow within our architecture passes different entities. Next, we present their individual responsibilities.

Trusted Sensors. All sensor data is protected from tampering (**R1**) through TEE-enabled and secure MMIO-supporting microcontrollers, e.g., Sancus. This guarantee is critical to provide end-to-end sensing. RFID readers, state-logging locks, IoT sensors such as temperature or humidity sensors, or cameras are suitable to capture physical information in our scenario. These sensors establish an attested connection to the TEE of the trusted server to ensure data authenticity (**R2**).

Trusted Server. Each company operates its own TEE-enabled trusted server, e.g., based on Intel SGX, that directly receives data from the trusted sensors via an attested and encrypted connection. Hence, we enable companies to trust sensed data to be authentic, even if it originated from an untrusted surrounding. All sensor data is processed inside a TEE to prevent any (external) tampering. Additionally, the server can detect inconsistencies such as dropped messages. The main task of this entity is to transform and persist attested sensor readings in a way that they remain verifiable in the future. To this end, the server pushes a *cryptographic fingerprint* of the data to a blockchain. This fingerprint cannot be altered and remains available as the blockchain is immutable and decentralized [5]. The raw data should be encrypted and retained either locally or at a dedicated (untrusted) storage provider to enable future verifiability. Here, we envision that most collaborators rely on cloud-based setups due to its virtually unlimited elasticity if excessive scalability needs surface.

Immutable Ledger. Once the fingerprints are on the blockchain, the trusted dataflow terminates as the sensed data is stored in a trustable immutable log. Related work [13] showed that cryptographic fingerprints offer reliable accountability. This entity thus enables companies to verify that claimed readings are correct on the one hand and match an attested checkpoint on the other hand. Hence, we introduce complete, trustworthy verifiability (**R3**) as long as the raw data is retained. Given that we only publish fingerprints, we do not have to make any sensitive information (publicly) available by default. While our architecture is oblivious of concrete instantiations, the ledger should not solely be maintained by collaborators as they might collude to manipulate data. For example, the participation of governmental surveyors or organizations working in the public's interest come to mind.

Storage Provider. A dedicated storage relieves the immutable ledger from potentially extensive storage needs. Access to this data is necessary in case of disputes or for future analyses of the supply chain. Each company is responsible for reliable data retention. If data is missing, we consider the company misbehaving (and liable for potential compensation). To account for privacy needs, this raw data should be encrypted by the trusted server [4]. Besides, the ledger could also record the granting of data access for transparency.

Next, we elaborate on challenges in real-world deployments.

C. Deployment Considerations of Our E2E Sensing

While the general design of our secure E2E sensing with its trusted dataflows and attested checkpoints is simple, we also have to discuss its deployment and operational influences. Our choice to rely on cheap, TEE-backed hardware not only improves our design's scalability but also makes the system deployable and retrofittable to existing supply chains (**R4**).

Transmission Gateway. As detailed in Figure 3, we can enhance our architecture with an optional (untrusted) transmission gateway that collects data from multiple sensors and sends it as batches to the trusted server. As such, it can relieve the sensors from energy-intensive tasks and reduce the hardware costs of each sensor node. This feature is especially desirable for continual condition monitoring during transit, i.e., the gateway can buffer messages to cope with offline phases. We deliberately do not set time constraints so that transmissions can occur once a (reliable) network connection is available. An absence of data can either be noticed by the trusted server if it does not receive all numbered sensor readings sent by the trusted sensors, or by the next party in the supply chain if the transmission gateway has been tampered with.

Verifiability of Computations. The trusted server can aggregate or filter any data before pushing it onto the ledger to persistently record digests as required by specific use cases. For example, instead of continual temperature readings, the trusted server can simply publish an aggregation of an intended range for a specific shipment. Due to the attestation between server and sensors, the server can rely on any aggregation that occurred on the trusted sensor nodes. Subsequently, any aggregation on the trusted servers will be attested and is verifiable for any recipient due to the properties of SGX attestations, i.e., all processing steps are publicly verifiable.

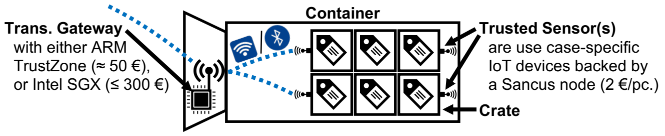

Fig. 3. A feasible deployment may consist of multiple crates equipped with cost-effective Sancus nodes. Optionally, a single transmission gateway per container can energy-efficiently relay messages to trusted servers in the cloud.

Detecting Mistakes. We require parties along the supply chain to visually check crates and to verify previously reported interactions, i.e., attested checkpoints, to enable an early detection of issues. Trusted sensors can facilitate the automation of this step. For example, each party scans the RFID tags on all crates to ensure that no previously reported crate is missing. If discrepancies arise, the previous hop must have withheld goods as long as the container is intact. Here, the verifying company is incentivized not to cover up any incidents as the next hop is likely to file a corresponding report to deflect blame. This approach ensures that incidents are correctly reported with a high probability, while only inflicting low verification overheads. In the absence of inconsistencies, attested checkpoints allow the system to prove supplier-introduced mistakes in received shipments. In this paper, we consider the exact conflict resolution mechanisms as out of scope.

Individual security and privacy needs can be addressed within our design as part of specific use case requirements.

V. Feasibility Study of E2E Sensing

We proposed E2E sensing with trusted dataflows for supply chains that provides companies with attested checkpoints. While trusted hardware addresses **R1** and **R2**, the ledger satisfies **R3**. To verify compliance with **R4**, we now present a preliminary performance evaluation and a cost analysis. Finally, we discuss security considerations for our solution as well as additional physical attack vectors.

A. Estimated Performance of our Trustworthy Architecture

At the level of servers or data centers, applications can be scaled with established means of industrial practice. Thus, the performance of TEEs in a distributed sensing application predominantly depends on the processing requirements and the data throughput on low-end sensing hardware.

Processing Rates. Sancus, a low-cost TEE, is built upon a lightweight 16-bit processor family that runs at 20 MHz clock speed, which restricts embedded applications to 65 kB of code and data, and relatively simple processing tasks [12]. The embedded cryptographic extensions of Sancus can process and transmit up to 200 kB/s. These characteristics allow a secure processing of sensor inputs, such as human inputs at IoT nodes, readings from temperature or humidity sensors, or queries of an RFID or GPS module, at a rate of several readings and transmissions per second [14]. More complex inputs, such as image or audio data, may take multiple seconds to process. As Sancus does not support public-key cryptography, the remote data center must manage the required symmetric keys. These restrictions come with the well-suited benefits for supply chains of an extremely low price below 2 €, and an operation time of several years on battery power.

Powerful Setups. Setups with comparable security guarantees could also involve microcontrollers with ARM TrustZone or even embedded PCs with Intel SGX. While they easily fulfill any processing requirements, they increase the per-unit prices (30–200 €) as well as the power consumption. Their needs might render a long-term battery operation impossible.

Scalability. Since every company maintains its own trusted servers that process data, our architecture scales dynamically. In particular, companies can utilize cloud computing to quickly scale out their processing capabilities when required.

B. Cost Analysis of Equipping Existing Supply Chains

To assess the deployment and equipment costs, we concretize our target scenario (cf. Section III-A). In Figure 3, we illustrate this detailed setup with multiple crates per container.

Crates. A typical crate could require up to five sensors: One, that detects an opening and closing of the crate, two humidity detectors, and two temperature sensors. Given their low bandwidth needs, a single Sancus for 2 € a piece would be sufficient to sample them at regular intervals (temperature and humidity) or on interrupt (open/close), and process or store input events for several hours. Realistically, this setup, including the basic sensors, can be built for less than 10 €. A prototype with off-the-shelf components may cost around 100 € and should be able to run on battery for several months.

Containers. Crates could rely on a gateway in close proximity, e.g., in the surrounding container or transport vehicle, which provides short-range connectivity to sensors (cf. Section IV-C). It further relays authenticated and (optionally) encrypted data to a trusted server within a data center. A TEE-enabled gateway could even provide additional tasks such as location tracking or scanning RFID tags when crates enter or leave the device's proximity. The price of such a system is hard to estimate as it depends on the communication and sensing equipment in use (e.g., satellite communication vs. GSM). The processing requirements can be satisfied with a TrustZone-enabled ARM microcontroller for 30 €. The overall price of a container's hardware could thus accumulate to around 300 €.

Data Center(s). Container and crate equipment would rely on extended storage and processing capabilities in a data center or in the cloud [4]. Data processing is handled by trusted execution environments to securely manage communication keys and to prevent tampering. Such SGX-enabled infrastructure is commercially available at marginally higher prices than today's commonly-used cloud infrastructures.

Miscellaneous. In this high-level analysis, we cannot account for use case-specific costs, e.g., highly precise humidity sensors. Furthermore, TEE attestation is based on cryptographic keys that are unforgably embedded into the deployed devices. In larger deployments, the secure deployment, storage, management, and revocation of keys need to be carefully designed [14], which implies further, unaccounted costs.

C. Security Discussion

The security considerations for our approach largely follow previous work [7] that aims for strong security guarantees in heterogeneous TEE deployments. Similarly, we follow a notion of security where all checkpoints that are recorded on the immutable ledger can be expressed with (i) the combined base of TEE hardware and application software of trusted sensors and servers, and (ii) the observed physical input events.

This notion means that all events recorded on the immutable ledger originate from a physical event. A core element in this context is the attestation between sensor nodes and a trusted server. We furthermore consider the lack of data availability and external attacks to be relevant for the system's security.

Attestation. All data acquisition along the supply chain is performed by software that is executed in TEEs with exclusive access to their respective sensors. The processing of this data occurs in attested enclaves that receive data from such sensor nodes. Since successful attestation uniquely binds the execution of an enclave to a trusted sensor, the authenticity of sensor readings can be guaranteed. If an enclave at the trusted server can establish the authenticity and freshness of an interaction that is being received, this data must originate from an associated trusted sensor. The backend enclave, i.e., the trusted server, can then persist the attested checkpoint immutably, which enables all parties to obtain an authenticity guarantee for the recorded interaction by attesting that enclave.

Lack of Data Availability. Sensor readings may be dropped during transit to cover up misbehavior. At the same time, transmission gateways must be allowed to cache data in case of offline phases. While the responsible trusted server will notice data gaps, malicious behavior should ideally be distinguishable from technical faults. The only technical solution to this problem is to make the gap in data transparent to the immutable ledger, which shifts the responsibility of investigating data gaps to the involved stakeholders in the system as an automatic analysis cannot distinguish malicious intent from faults.

External Attacks. Prior work [15] showed that software vulnerabilities in software can extract secure data even from TEEs. However, we consider these software vulnerabilities as an orthogonal problem as existing best practices, such as software verification and regular updates, should be maintained to fully mitigate this issue in a deployed production system.

D. Open Physical Attack Vectors of our Architecture

Apart from the security aspects of our architecture, we also have to consider the boundaries of our E2E sensing design.

Attacks on Sensors. Unfortunately, no technical solution can prevent direct tampering with the sensors. This situation includes physical attacks (e.g., placing the sensors in a manipulated environment that differs from the one to be reported in), but also human-induced inconsistencies (e.g., registering the wrong sensor). We envision that both cases would have to be dealt with by successive parties when scanning and visually inspecting the correctness. Additionally, physical attacks on the sensor and deployed nodes are known issues that have to be mitigated through orthogonal research [16], [17].

Physical Linking of Products. The scrutiny against integrating TEEs into supply chain processes stems from the lack of solutions to reliably interconnect the physical and the digital world [18]. To this end, recent advances propose to rely on new trust anchors to link a product to blockchain data to ensure their authenticity and to fight counterfeit products [19]. Related research [20], [21] looks into the reliable identification of workpieces. To increase trust, suppliers require ways to attach markers to products, batches of products, and containers.

VI. Related Work

Supply chain research is a large research area. Thus, we focus on work that researched the utility of distributed ledgers for supply chains. Work in other domains already considered the use of TEEs to immutably persist data in blockchains.

Supply Chains and Blockchain. Initial blockchain-backed applications seize the opportunity to remove trusted third parties from inherently distributed processes [18]. Other directions look into asset recording, e.g., to identify counterfeit products [22] or to promote fair trade [23]. Blockchains can further help to record trade events as well as help to improve the overall verifiability and accountability [1]. Similarly, research looked into tracking and tracing of products [1], [24]. However, these approaches only consider how to persist data untampered (**R1**), but fail to consider data authenticity (**R2**).

Trusted Computing and Blockchain. Microsoft proposed the TEE-backed Confidential Consortium Framework (CCF) [25] that utilizes trusted computing to secure deployed services. In particular, a TEE-backed key-value store enables a permissioned distributed ledger to run on multiple TEEs. This design allows participants to also trust the executed software on the other machines, but is inapplicable to our scenario (**R4**).

Prior work [18] raised concerns about tamperproof sensors in connection with supply chains and blockchains with the detachment between trusted sensors and reported data as their main concern. We provide more insights into this aspect (**R3**) by moving the trust in our E2E sensing to the edge of the supply chain environment and correct this misconception through our trusted dataflows. In our design, we rely on trusted sensors to report any interaction with products. To the best of our knowledge, we show this feasibility for the first time.

VII. Conclusion and Future Work

We presented an architecture to address a lack of trusted sensing in supply chains and the monitoring of interactions and conditions. By using trusted hardware in the form of trusted execution environments along the complete dataflow of sensed data, we establish a trusted dataflow from a trusted sensor to a long-term storage and a distributed ledger that persistently records sensor readings and fingerprints thereof for future verifiability. With attested checkpointing, we are also able to attest events even if they occurred in otherwise untrusted surroundings. Based on our feasibility study, i.e., security discussion and preliminary evaluation, we conclude that our system addresses all derived requirements (cf. Section III-B).

In alignment with related work (cf. Section V-D), we expect that significant effort is required in future work to develop reliable solutions for scenarios where an embedding or attachment of trust anchors to a product is impossible or infeasible (e.g., for cost or practicality reasons). Furthermore, we have to investigate the risks of attacks at the boundaries of our architecture in more detail. Finally, we envision to conduct a real-world evaluation to verify our derived feasibility claims.

Acknowledgments

This work is funded in parts by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC-2023 Internet of Production – 390621612. This research is partially funded by the Research Fund KU Leuven. Fritz Alder is supported by a grant of the Research Foundation – Flanders (FWO).

References

[1] V. Dedeoglu *et al.*, "A Journey in Applying Blockchain for Cyberphysical Systems," in *COMSNETS*, 2020.

[2] J. Pennekamp *et al.*, "Towards an Infrastructure Enabling the Internet of Production," in *IEEE ICPS*, 2019.

[3] J. Pennekamp *et al.*, "Dataflow Challenges in an *Internet* of Production: A Security & Privacy Perspective," in *ACM CPS-SPC*, 2019.

[4] M. Henze, "The Quest for Secure and Privacy-preserving Cloud-based Industrial Cooperation," in *IEEE SPC*, 2020.

[5] J. Pennekamp *et al.*, "The Road to Accountable and Dependable Manufacturing," *Computer*, 2020.

[6] S. Pinto *et al.*, "IIoTEED: An Enhanced, Trusted Execution Environment for Industrial IoT Edge Devices," *IEEE Internet Comput.*, vol. 21, 2017.

[7] J. Noorman *et al.*, "Authentic Execution of Distributed Event-Driven Applications with a Small TCB," in *STM*, 2017.

[8] P. Maene *et al.*, "Hardware-Based Trusted Computing Architectures for Isolation and Attestation," *IEEE Trans. Comput.*, vol. 67, no. 3, 2017.

[9] Z. Zheng *et al.*, "An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends," in *IEEE BigData Congress*, 2017.

[10] F. McKeen *et al.*, "Innovative instructions and software model for isolated execution," in *HASP*, 2013.

[11] S. Pinto and N. Santos, "Demystifying Arm TrustZone: A Comprehensive Survey," *ACM Comput. Surv.*, vol. 51, no. 6, 2019.

[12] J. Noorman *et al.*, "Sancus 2.0: A Low-Cost Security Architecture for IoT Devices," *ACM Trans. Priv. Secur.*, vol. 20, no. 3, 2017.

[13] G. Zyskind *et al.*, "Decentralizing Privacy: Using Blockchain to Protect Personal Data," in *IEEE SPW*, 2015.

[14] J. Van Bulck *et al.*, "VulCAN: Efficient Component Authentication and Software Isolation for Automotive Control Networks," in *ACSAC*, 2017.

[15] J. Van Bulck *et al.*, "A Tale of Two Worlds: Assessing the Vulnerability of Enclave Shielding Runtimes," in *ACM CCS*, 2019.

[16] R. Anderson and M. Kuhn, "Low Cost Attacks on Tamper Resistant Devices," in *Security Protocols*, 1997.

[17] P. Kocher *et al.*, "Differential Power Analysis," in *CRYPTO*, 1999.

[18] K. Wüst and A. Gervais, "Do you need a Blockchain?" in *CVCBT*, 2018.

[19] IBM Research, "Combating fraud with blockchain and crypto-anchors," https://www.research.ibm.com/5-in-5/crypto-anchors-and-blockchain/, 2020 (accessed April 1, 2020).

[20] S. Pollard *et al.*, "Authentication of 3D Printed Parts using 3D Physical Signatures," in *NIP & Digital Fabrication Conference*, 2018.

[21] D. M. S. Velandia *et al.*, "Towards industrial internet of things: Crankshaft monitoring, traceability and tracking using RFID," *Robot. Comput. Integr. Manuf.*, vol. 41, 2016.

[22] Everledger Limited, "Diamonds," https://www.everledger.io/industry-solutions/diamonds/, 2020 (accessed April 1, 2020).

[23] S. A. Abeyratne and R. Monfared, "Blockchain ready manufacturing supply chain using distributed ledger," *Int. J. Res. Eng. Technol.*, vol. 5, no. 9, 2016.

[24] J. Pennekamp *et al.*, "Private Multi-Hop Accountability for Supply Chains," in *BIoTCPS (ICC Workshops)*, 2020.

[25] M. Russinovich *et al.*, "CCF: A Framework for Building Confidential Verifiable Replicated Services," Microsoft, Tech. Rep., 2019.