Blockchain-Based Privacy Preservation for Supply Chains Supporting Lightweight Multi-Hop Information Accountability

Lennart Bader[♦,a,b], Jan Pennekamp[♦,a,*], Roman Matzutt[a],
David Hedderich[c], Markus Kowalski[c], Volker Lücken[c], Klaus Wehrle[a]

[a] *Communication and Distributed Systems, RWTH Aachen University, Aachen, Germany*
[b] *Cyber Analysis & Defense, Fraunhofer FKIE, Wachtberg, Germany*
[c] *e.GO Mobile AG, Aachen, Germany*

Abstract

The benefits of information sharing along supply chains are well known for improving productivity and reducing costs. However, with the shift towards more dynamic and flexible supply chains, privacy concerns severely challenge the required information retrieval. A lack of trust between the different involved stakeholders inhibits advanced, multi-hop information flows, as valuable information for tracking and tracing products and parts is either unavailable or only retained locally. Our extensive *literature review* of previous approaches shows that these needs for cross-company information retrieval are widely acknowledged, but related work currently only addresses them insufficiently. To overcome these concerns, we present PrivAccIChain, a secure, privacy-preserving architecture for improving the *multi-hop information retrieval with stakeholder accountability* along supply chains. To address use case-specific needs, we particularly introduce an adaptable configuration of transparency and data privacy within our design. Hence, we enable the benefits of information sharing as well as multi-hop *tracking and tracing* even in supply chains that include mutually distrusting stakeholders. We evaluate the performance of PrivAccIChain and demonstrate its real-world feasibility based on the information of a purchasable automobile, the *e.GO Life*. We further conduct an in-depth security analysis and propose tunable mitigations against common attacks. As such, we attest PrivAccIChain's practicability for information management even in complex supply chains with flexible and dynamic business relationships.

Keywords: multi-hop collaboration, tracking and tracing, Internet of Production, e.GO, attribute-based encryption

1. Introduction

Supply chain management (SCM) relies on large-scale information systems for tracking the relevant activities of numerous, potentially globally distributed, suppliers [1]. By definition [2], supply chains involve multiple steps where each hop corresponds to handing over goods between different companies, i.e., different stakeholders. All goods, parts, or intermediate products are passed along the supply chain with intermediate production and assembly steps until, eventually, the final product is assembled [3]. Obtaining sufficient information about each step is crucial for the final producer to enable tracing parts either for quality assurance or to assert a sustainable or otherwise preferable origin of the product's raw materials [4]. While the ongoing digitalization of production processes benefits generating the required data, information retrieval becomes more challenging as more fine-grained data is being generated in a highly distributed manner [5]. Additionally, widely gathered feedback data, e.g., information on a customer's usage of the product [6], further increases the complexity of desired information retrieval from SCM systems.

While traditional SCM mainly deals with the timely shipment of goods, meeting demand with sufficient quantity, and advanced storage management [3, 7], the increasing digitalization and additional requirements for information retrieval along the supply chain highlight the need for more holistic SCM approaches. Besides handling the vastly

*Corresponding author. Tel: +49-241-80-21411, Fax: +49-241-80-22222
Postal address: Informatik 4 (COMSYS), RWTH Aachen University, Ahornstr. 55, 52074 Aachen, Germany
Email addresses: {bader, pennekamp, matzutt, wehrle}@comsys.rwth-aachen.de
♦Equal Contribution.

Authors' version of a manuscript that was accepted for publication in *Information Processing & Management*. Changes may have been made to this work since it was submitted for publication. Please cite the published version:
https://doi.org/10.1016/j.ipm.2021.102529

increased volume of relevant data gathered at each step, we also must rethink collaboration strategies for different stakeholders. Established localized profit optimizations for individual suppliers can lead to miscalculations regarding demand and available supplies, transportation capabilities and delays, as well as general logistic problems, such as deliveries to unintended locations, inappropriate packaging, or incorrect labeling [7–9]. The costs resulting from these (potentially avoidable) miscalculations and mistakes quadruplicated over a period of 24 years for certain industries [10]. To counteract this development, the joining of business operations across company boundaries as well as a sharing of profits and risks within the complete supply chain has been proven to increase trust between business partners, to accelerate business performance, and, ultimately, to improve customer satisfaction and competitiveness [11, 12]. Transitioning to an information management that allows for supply chain-wide information retrieval despite the potential lack of trust between some participating suppliers is crucial to seize these currently unrealized potentials.

These requirements are further expressed in the recent notation of the Internet of Production (IoP) [6, 13], where the selection of suppliers is assumed to be highly dynamic based on momentary needs. This collaboration model further exacerbates the need for fine-grained and trustworthy information retrieval along supply chains. In addition to apparent trust issues among previously unaffiliated companies [7], the localized information storage, i.e., compartmentalization, remains a major open challenge [14, 15]. Namely, producers remain unable to validate claims by their suppliers due to a lack of sharing crucial information, and keeping sensitive information in centralized data vaults requires appropriate (data) security measures [16], especially considering the increasingly volatile context of the IoP [16].

Even though research and industry are aware of these issues [3, 7, 17, 18], newly proposed approaches do not offer satisfactory solutions, as they lack appropriate considerations of data privacy [19–25], multi-hop environments [26–28], or scalability [23, 29, 30]. Yet, related work underpins the relevance of our outlined scenario [31] and the resulting use cases [32, 33]. Especially solutions to track and trace products and goods along the supply chain [31] allow for a timely and reliable identification of and reaction to issues within the supply chain. Furthermore, establishing collaboration without prior trust relationships requires to provide stakeholders with advanced accountability and verifiability features [34]. Blockchain technology, an emerging solution in various domains (e.g., health care [35], manufacturing [36], or smart cities [37]), has previously been identified as a valuable candidate [3, 38, 39] to offer these features via its decentralized and immutable event log to achieve consensus even among mutually distrusting parties.

By taking blockchain technology and previously proposed approaches into account, we provide a fully-featured supply chain architecture that especially addresses multi-hop information flows while also providing accountability. To account for the sensitivity of (shared) information as well, we consider the trade-off between privacy and transparency, i.e., we provide an adaptable design to realize the envisioned benefits without introducing unwarranted privacy risks.

Contributions. In this paper, we utilize blockchain technology to improve the information availability and accountability across multiple hops of large supply chains while carefully gauging data privacy and transparency. This paper extends our previous work [40] that proposes the initial architecture for providing multi-hop accountability along supply chains. We extend our work by systematically reviewing related work, detailing all design aspects of our proposed architecture, *PrivAccIChain*, and by thoroughly evaluating PrivAccIChain based on a real-world supply chain of an in-store purchasable automobile, the e.GO Life. Overall, we make the following contributions in this paper.

- We provide an in-depth literature review of existing approaches that propose approaches for multi-hop information retrieval along supply chains. We investigate both, centralized and blockchain-backed, approaches based on ten comprehensive system properties (cf. Table 1) and identify the need for further research in this domain.
- We present the full details of our novel and previously outlined [40] architecture, *PrivAccIChain*, that achieves a tunable trade-off between data privacy and transparency. Namely, we provide additional details on our hybrid encryption scheme, the required policy design, and the blockchain-based verification of recorded fingerprints.
- We base our performance evaluation on a real-world use case to showcase PrivAccIChain's applicability by decomposing the supply chain of a purchasable automobile, which consists of an assembly of over 2000 parts.
- To address potential concerns in industry, we cover twelve potential attack vectors against PrivAccIChain in our security discussion. We discuss their likelihood and severity and provide suitable and tunable countermeasures.

Paper Organization. In Section 2, we detail our considered scenario and emphasize the need for information retrieval along supply chains, i.e., (multi-hop) tracking and tracing. In Section 3, we compare existing approaches based on ten comprehensive properties. Then, we derive the design goals of suitable architectures in Section 4 before presenting PrivAccIChain in Section 5. In Section 6, we analyze the real-world supply chain of a car manufacturer for our evaluation in Section 7. In Section 8, we discuss the security of our approach, before concluding in Section 9.

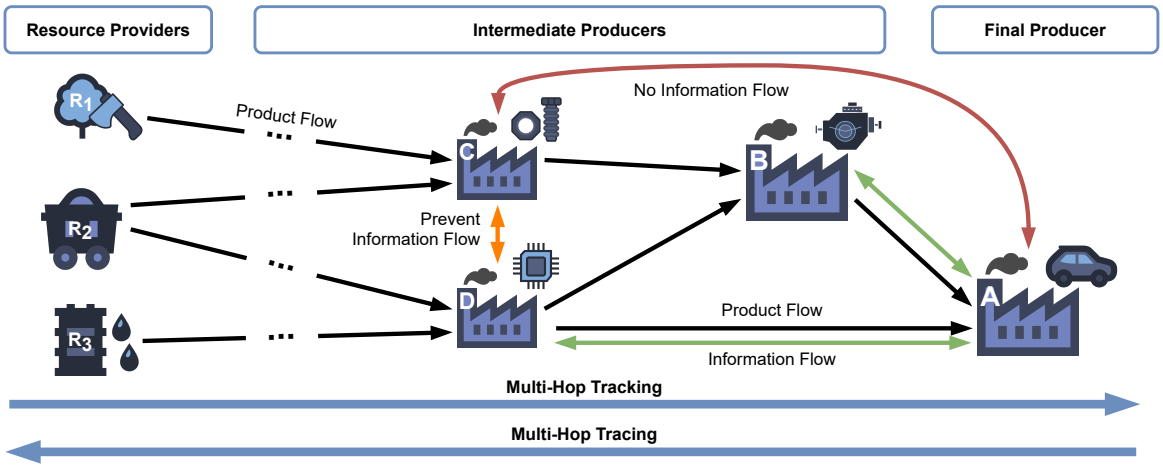


Figure 1: In our scenario, we consider the complete supply chain from the resource provider (e.g., from the oil field $\texttt{R}_3$), over possibly multiple intermediate producers, to a final producer `A` assembling or manufacturing the final product. Today, information flows between direct partners are well-established [41]. However, information flows along the horizontal dimension are not widely in place even though they would enable multi-hop tracking and tracing [7]. Vertical information flows require special consideration as they entail further, potentially unwanted, consequences [42].

2. Considered Supply Chains and their Desirable Information Flows

Due to the varying definitions of supply chains in related work, we first present our view as a foundation for the remainder of the paper. To this end, we define the structure of supply chains in Section 2.1, discuss existing information flows in Section 2.2, and detail the need for tracking and tracing in Section 2.3. Finally, we derive six desirable properties of approaches that attempt to improve the information flows and the data sharing along supply chains. In Section 3, we partially base our analysis of existing work on this set of desirable properties.

2.1. The Structure of Today's Supply Chains

Supply chains consist of multiple hops, each representing a company that serves as a supplier, producer, customer, or service provider, such as transportation. Overall, they originate from an initial resource and lead to a final product. While the business relationships of these companies are bidirectional (e.g., goods flow in one direction and payments in the other direction) and can be represented as an undirected graph, product flows always constitute an acyclic process, i.e., product flows correspond to a directed acyclic graph (DAG). In Figure 1, we visualize an exemplary supply chain graph that consists of resource providers, represented by nodes with no incoming edges, intermediate producers that have both incoming and outgoing edges, and a single final producer with only having incoming edges.

Different product flows are combined by production processes, i.e., we describe them with nodes in the DAG. We further consider ownership transfers, which we detail with edges in our considered DAG. Hereby, we consider transportation services as separate production processes, i.e., we model each product flow with only two different actions: namely a *produce* step and a *trade* step, which is later incorporated into our data record structure (cf. Section 5.5). Besides product flows, the DAG also represents the product composition by linking the respective flows to each other. This linkage, in theory, allows involved companies to determine products that utilize certain subcomponents as well as to identify those components that are used by a composed product by traversing the DAG for information retrieval.

Following this description of physical flows in our considered supply chain scenario, we next introduce existing (digital) information flows along supply chains in more detail.

2.2. Information Flows between Business Partners and Suppliers

Apart from physical product flows, modern supply chains also include (digital) information flows between business partners. Nowadays, information usually only flows between direct partners and, as a result, the flow of information over multiple hops is only limited, possibly severely delayed or non-existing. With respect to our supply chain example

in Figure 1, an information flow between the Companies A and B exists. An exchange of information between Producer A and Producer C, however, is usually missing, even though such an advanced multi-hop information flows would, for instance, enable common use cases such as tracking and tracing of individual products [31], and joint operations between indirect business partners [11]. While a collaboration of businesses over multiple hops has been identified to improve the business performance by allowing faster responses to supply shortcomings or changed customer demands [13] as well as inter-business production optimizations [43], a revelation of business secrets can negatively impact the competitiveness [7]. Therefore, companies currently limit the data sharing to long-term business relationships with well-established trust [7], i.e., dynamic and short-lived relationships are severely inferior in this regard.

Low-Trust Relationships. In contrast to these well-established relationships, we explicitly consider short-term business relationships that are quickly established and potentially short-lived with the purpose of adapting to (temporarily) changed requirements and needs. Hence, they lack the valuable long-term trust and, therefore, impede the flow of information. For such low-trust environments, traditional approaches for information-based collaboration and information retrieval are unsuitable as they require mutual trust and do not take these novel low-trust relationships into account. Naturally, traditional business relationships should also be supported in such settings. To offer significant benefits through multi-hop information sharing in combination with low-trust business relationships that do not negatively impact the companies' competitiveness, careful decisions on which information should be shared are required. These decisions heavily depend on the considered use case and the involved stakeholders. For example, in supply chains with low volumes, part numbers can reveal interesting details on the supplier's production capabilities and utilization [14], while they are completely insensitive for mass-produced articles. In addition, the company that shares the data should remain in control of granting access to the information. Consequentially, a potential central entity that handles these data exchanges or relays information must implement appropriate access control.

Referencing the scenario that we outline in Figure 1, Company C and Company D neither have a direct nor multi-hop business relationship in our considered supply chain. Therefore, information flowing between these parties is usually undesired and should not be fostered or even promoted. However, information flows between the Companies C and A should be supported even though their business relationship is only indirect. The sharing of information on production processes and product properties particularly yields the applications of tracking and tracing along the supply chain. To further elaborate on these applications, we explicitly look into the identification of production and product issues over multiple hops in the following.

2.3. The Benefits of Multi-Hop Tracking and Tracing

We now introduce tracking and tracing as specific, but common [31] use cases for multi-hop information flows in digitized supply chains that offer a benefit especially in volatile and short-lived supply chains, e.g., scenarios where companies dynamically react to customer change requests. Tracking and tracing allow companies to identify previous and subsequent production steps and the respective products over multiple hops. The ability to identify the root cause of a problem [44] as well as potentially affected subsequent products is relevant for several scenarios. Here, quality assurance [4] as well as the minimization of harm for customers and other businesses are prominent examples. Apart from recalls of tampered or spoiled food products [31], tracing of individual components is highly relevant following car or plane accidents [45], where the process of revealing product histories relies on manual inspections that require longer periods of time and cannot guarantee a successful outcome. Similar considerations also hold for valuable, safety-critical medical products with large quantities [46]. Due to the significance of tracking and tracing individual products and components, which we also underline in Section 6, we explicitly include these applications in our scenario.

Tracking. Tracking a product allows companies to identify subsequent products that utilize or modify the former. In Figure 1, tracking a microcontroller produced at Company D would either reveal the engine assembled at Producer B as well as the automobile at Company A over two hops, or just the automobile at A, depending on the product flow of the specific product instance. The possibility to identify subsequent products is beneficial for quality assurance, for instance, since tracking allows companies to identify those products that are potentially affected by a faulty production charge of a specific product.

Tracing. Similar to tracking, tracing represents the process of identifying subcomponents of a product. In the example in Figure 1, tracing based on the final good (automobile, A) reveals the following previous production steps and the corresponding products. The revelation covers the engine at B and the microcontroller at D, but also the screws at C and all previous production steps until the resource providers are revealed. As for tracking, the provision of product information over multiple hops is the crucial requirement for automated, reliable and fast tracing that allows both the

revelation of full supply chain structures as well as dedicated inspection of single production paths. Due to the potential of gaining advanced insights into existing supply chain structures, considerations for enabling tracing always need to cover respective privacy preservation mechanisms.

Combining both, tracking and tracing, allows for advanced multi-hop product analyses as well as sophisticated quality control [38, 44]. We exemplary consider the case where Company C detects an issue with multiple products that is caused by one or multiple subcomponents. Provided that a specific charge produced by the metal provider R_2 is identified as the root of the problem with the help of tracking, C can provide these insights to R_2 as well as the downstream multi-hop business partners B and A, who can react to the issue early on. Furthermore, R_2 can utilize tracing to identify those companies that might be affected by the same issue, e.g., D and, iteratively, B and A. Hence, the combination of tracking and tracing allows businesses to identify the root of production issues and to reduce follow up costs by informing direct and indirect business partners on these issues at an early stage. The identification of production issues and the subsequent determination of affected products positively affects the trust of customers and business partners, allows for improved lifetime estimates, reduces the need for maintenance downtimes, and further reduces costs resulting from undetected product issues. However, careful access control mechanisms are necessary to minimize unintended data revelation or even data leaks that might have a negative impact on the company's competitiveness [7, 47].

Based on the outlined supply chain scenario and tracking and tracing as prominent and important applications, we derive a set of desired properties for multi-hop information sharing in today's and tomorrow's digitized supply chains.

2.4. Desirable Properties for Approaches Improving the Data Sharing Along Supply Chains

We identify the following six properties that address collaboration within our previously presented scenario (cf. Section 2.1). We consider these properties to be an essential foundation for respective approaches, since the combination of low-trust business environments and the presented applications directly yield these properties.

P1: No-Trust Assumption. An approach should not require trust between different participants for its operation. In particular, a limited amount of trust is only needed for direct or indirect business relations. Companies that do not have business relationships do not have to trust each other at all. Since our scenario explicitly considers short-term business relationships and the matching lack-of-trust environments, approaches that require well-established trust relationships. Thus, only approaches that take the no-trust assumption into account can provide benefits for dynamic supply chains.

P2: Data Privacy. Following the limited trust, we determine advanced data privacy features as crucial, i.e., approaches must consider the companies' data privacy and provide privacy-preserving methods such that businesses can protect their information from unauthorized access. In particular, appropriate data encryption in combination with access control mechanisms is critical. Otherwise, businesses will isolate their data, i.e., prevent multi-hop data sharing.

P3: Accountability. All approaches should provide accountability features for processed data, i.e., it provides guarantees on data existence and protection against manipulation, and does not only serve as a storage location for unverifiable and illegitimate data. Due to the potential short-term business relationships, those sophisticated guarantees are essential for achieving any benefits from utilizing the respective architecture.

P4: Multi-Hop Capabilities. Practical approaches must break the multi-hop barrier between different business to provide a global view on supply chain data and its product flows. Since we explicitly focus on multi-hop collaboration and tracking and tracing as popular applications, any approach must reliably enable multi-hop information flows.

P5: Product Modeling. Suitable approaches model the supply chain, the production processes, and products. Thus, they can handle information on individual products and related production steps. This data ultimately helps companies to identify production issues and to assess the quality of individual products. Utilizing multi-hop data and information flows, product modeling further enables autonomous multi-hop tracking and tracing.

P6: Scalability. Proposed designs must offer a scalable concept to support large-scale, real-world supply chains. To introduce benefits to supply chains, every architecture needs to adequately scale with the number of participating companies as well as the amount of shared information (number of product flows and volume of information flows). The latter factor is influenced by both the production volumes of involved companies and the covered time period of supply chain interactions.

In the next section, we present a survey on approaches that (remotely) address our scenario, as these approaches consider product digitization, tracking and tracing, and multi-hop information sharing in general. Based on this survey, we further extend our set of desirable properties and compare the covered approaches in Section 3.3.

3. Literature Review on Related Work in Supply Chains with Multi-Hop Information Sharing

In this section, we conduct an extensive review of previous academic literature on digital collaboration in supply chains as a foundation for our work. In Section 3.1, we first present our methodology for considering related work in our review and present an overview of existing surveys in this research area. Subsequently, we provide a high-level overview of existing approaches in Section 3.2. Finally, we extend our list of desirable properties to a comprehensive list (**P1**–**P10**) for respective supply chain platforms and compare all presented approaches accordingly in Section 3.3.

3.1. The Methodology of our Literature Review and Related Surveys

Even though blockchain technology is still a comparably new building block for decentralized services, its advantages for supply chain environments are already well-acclaimed. As a consequence, several surveys [31, 48–55] study the use of blockchain technology regarding its applicability in the context of supply chain management. Other related surveys look into the impact of blockchains on the industrial sector as a whole [56], on specific industries (e.g., construction [57] or pharmaceuticals [58]), or logistics in general [28]. A common understanding is that open adoption challenges exist (e.g., Gonczol et al. [31]) and the use of blockchain technology also introduces constraints (e.g., Hald and Kinra [51]). Other publications motivate further promising research directions. For example, Wüst and Gervais [3] look into the general feasibility of applying blockchain technology and Malik et al. [59] offer interesting insights into blockchain-based reputation systems. Similarly, researchers advocate for targeted advances, for example, in logistics [34], pharmaceuticals [60], and finance [61].

In contrast to those previous surveys, our literature review focuses on multi-hop tracking and tracing capabilities in low-trust environments as we consider this aspect a crucial enabler for future use cases (cf. Section 2). Therefore, we only include approaches that (at least partially) have multi-hop tracking and tracing in mind. Consequentially, most considered approaches have been covered already by related surveys, but to the best of our knowledge, our literature review is the first to specifically investigate currently available multi-hop capabilities holistically. Gonczol et al. [31] recently provided an excellent first overview by dedicating special attention to approaches enabling product traceability. We extend upon their work by (i) considering further approaches that are not included in the survey conducted by Gonczol et al., and by (ii) assessing all approaches regarding our set of ten desirable properties for supply chain platforms considering multi-hop tracking and tracing. In particular, we cover the desirable properties that we identified before (cf. Section 2.4) as well as more general assessment properties (cf. Section 3.3). Additionally and in contrast to existing surveys, we further broaden the scope of our review and its impact by also considering centralized approaches (e.g., Appelhanz et al. [62]), i.e., we integrate non-blockchain-backed approaches into our analysis.

3.2. Existing Supply Chain Proposals and Implementations

We partition our further discussions according to the existing approaches' main deployment model, i.e., whether they operate in a centralized manner (Section 3.2.1) or based on a blockchain as a decentralized ledger (Section 3.2.2).

3.2.1. Centralized Platforms

As centralized platforms do not require any synchronization between the participating entities, they promise better performance than decentralized platforms. Furthermore, they ease the bootstrapping of new collaborators as all participants will connect to the centralized platform. Depending on the exact scenario, waiving the use of blockchain technology can entail better scalability (e.g., Nayak et al. [25]). Consequentially, in business environments with well-established trust relationships between direct and potentially even indirect business partners, centralized approaches offer a viable platform design. However, a fully centralized platform can neither provide advanced accountability features nor enhance trust among collaborators [63], such that (limited) trust between business partners is crucial for the applicability of such centralized platforms, i.e., they are usually unsuitable for tomorrow's dynamic low-trust settings.

Despite the requirement for pre-existing trust relations, centralized approaches can still achieve improved transparency between companies and customers over multiple hops. Appelhanz et al. [62] propose a traceability system for wood products that aims to increase customer's purchase intentions by providing information on eco-friendliness and general production conditions over multiple hops towards the customers [62]. To map products to digital representations, they consider Ink-printing as well as RFID tags, which are also common among others, not necessarily centralized, approaches [23, 64–67]. Further centralized approaches address similar challenges, such as food traceability [64, 65, 67, 68], textile and clothing supply chain management [25], and tracking of tools for construction

sites [69]. Although trust is explicitly required (e.g., Appelhanz et al. [62]) for certain approaches, such an assumption is viable for well-established business relations or approaches that serve as a digital information platform for easing work processes (e.g., Goodrum et al. [69]). Due to the centralized data management, accountability features can only be provided by the platform operator. However, the presence of stronger trust relations improves the overall scalability as the need for technical guarantees is reduced in this setting.

3.2.2. Blockchain-based Platforms

In contrast to centralized approaches, blockchain-based platforms avoid the necessity for trust between all collaborators by design, but consequentially face different challenges. Storing data immutably on a blockchain requires careful a design process regarding (i) what data to store and (ii) to select appropriate data volumes to manage on-chain. Otherwise, blockchain-based approaches are prone to permanent privacy breaches and poor scalability. However, a decentralized and tamperproof data storage creates accountability even in low-trust environments. Utilization of data encryption and offloading larger data amounts to off-chain storages can mitigate privacy and scalability concerns. Since on-chain data is irrevocable, the applied encryption schemes have to provide special resistance against common attacks as well as post-quantum security [70]. Application of these features requires careful considerations of the resulting trust implications, i.e., inappropriate designs can re-introduce undesired trust assumptions and requirements.

In the remainder of this section, we present and discuss existing approaches for enabling multi-hop collaboration along supply chains with the help of blockchain technology. Hereby, we focus on identifying their respective strengths and weaknesses regarding data privacy, scalability as well as the underlying trust assumptions and requirements.

Our previous line of arguments on the drawbacks and benefits of utilizing blockchain technology is backed by Abeyratne and Monfared [19]. They identify the value of blockchain technology for achieving consensus in low-trust environments, where the limited trust between companies as well as customers can be improved by enabling a tracing of products and by providing transparency for production processes. They further explicitly consider digital certificates to assert the desired product quality. Extending upon this proposal by further reducing paperwork using a blockchain can significantly reduce management costs resulting from manual accounting [34]. Similarly, Gao et al. [71] criticize that maintaining a global view on a supply chain is challenging due to the variety of involved systems. As a mitigation, they suggest to rely on blockchain technology to improve this situation.

Achieving appropriate scalability properties is a challenge for all blockchain-backed approaches in the context of supply chains, since large amounts of transactions lead to growing blockchain sizes and might hit the blockchain's limitations regarding its achievable transaction throughput. Especially those approaches that keep track of individual goods and their composition over multiple hops on-chain [20, 21, 23, 24, 29, 30, 38, 66, 72, 73] might suffer from poor scalability regarding the complexity of today's supply chain structures. However, specifically targeting more narrow use cases of underlying supply chains can suffice to maintain appropriate scalability. We observe examples regarding supply chains for food [18, 23, 24, 38, 66, 74–78], high-value goods such as jewelry or art [17, 63], or pharmaceuticals [27, 79–84], as well as considering logistics only for already assembled products, product batches [26, 28, 85–88] or the post supply chain [22]. The real-world applicability of blockchain-based approaches, in general, is underpinned by the growing number of production-grade platforms [17, 18, 27, 63, 72, 73, 79, 80, 82, 85, 87, 89]. Everledger [17], for instance, is a commercial project that provides accountable tracking and tracing of high-value goods, i.e., diamonds, gemstones, wine, art, and further luxury goods, similarly to Project Provenance [63]. Although Everledger is backed by a blockchain, it partially relies on a trusted third party (TTP). They address the drawbacks of relying on a TTP by requiring a certification after ISO 27001 [90], which attests appropriate information treatment.

Despite their scalability challenges [71], approaches that consider multiple hops of a supply chain can still provide promising performance, appropriate scalability, and comprehensive functionality. Malik et al. [38], for instance, propose ProductChain as an architecture for provenance in food supply chains that enables the blockchain-backed tracking and tracing of food products. They specifically consider the inclusion of sensor data, e.g., temperature measurements during transportation, as well as the encryption of this information, such that their approach addresses most of the desired properties (cf. Section 2.4). However, they assume a special trust among participants as participants have to be certified partners. Further, as tracking and tracing are implemented based on the information stored on the blockchain, the respective references between different products are opaque to all participants. The proposal of Wang et al. [20] for multi-hop tracking and tracing entails a similar privacy issue, as the respective references are not encrypted. Providing privacy-preserving features for such meta information as well either requires on-chain encryption of tracking and tracing references or storing the respective information off-chain. The latter concept is also proposed for larger payloads

by Weber et al. [91] for improved scalability. Instead of modeling individual products, they propose an accountable blockchain-backed business process monitoring and active mediation between untrusted business partners.

Besides the targeted use cases (e.g., food chains) and the decisions on required properties for these use cases, selecting the underlying blockchain technology further impacts the applicability of any approach in real-world deployments. While public ledgers, e.g., the public Ethereum network, promise strong, public accountability, permissioned ledgers with a dedicated purpose, i.e., a specific supply chain, offer advanced features, higher transaction throughputs and better scalability [92]. These considerations are in accordance with guidelines on the applicability of blockchain for different use cases, including digitized supply chain environments, as discussed by Wüst and Gervais [3]. They argue that the utilization of blockchain technology is only vindicated in low-trust supply chain environments, and only permissioned ledgers provide additional value for the associated use cases.

Hereinafter, we extend our list of desirable properties (cf. Section 2.4) to also cover (technical) readiness and compare the discussed approaches as well as our previous work [40] with respect to the resulting set of ten properties.

3.3. Comparison of Existing Work

To provide a standardized comparison and an overview of related work regarding our scenario (cf. Section 2), we rate the fulfillment of the desired properties (cf. Section 2.4) for all previously mentioned approaches. In addition to those desired properties, we assess the *readiness* of each approach by considering the following for additional and scenario-independent properties, i.e., we give pointers regarding their potential impact in real-world deployments.

P7: Implementation. To assess whether an approach is ready for real-world deployment, we consider the existence of (prototypic) implementations beyond the presented concepts. Additionally, we also consider the public availability of claimed implementations, with a special focus on open-source releases. Publicly available implementations are favorable, as they allow for an independent evaluation of any claimed properties and an independent evaluation.

P8: Evaluation. Given that the performance is a crucial aspect when judging the real-world feasibility, we also rate whether the performance and applicability have been evaluated comprehensively, i.e., we check whether they are based on a real-world scenario and whether they are based on realistic assumptions for their conducted evaluation.

P9: Universality. To allow for a fair comparison between use case-specific and universal approaches, we also indicate whether the respective approach is universally usable, i.e., the approach should also operate in differently structured supply chains and should not be bound to a single use case or product.

P10: Blockchain Type. As we detailed in Section 3.2, existing approaches are either centralized or blockchain-backed. From now on, we differentiate between the blockchain deployment models of blockchain-backed approaches.

With this selection of ten properties, we can now provide a holistic assessment of existing approaches that facilitate digitized collaboration in supply chains. We summarize our findings in Table 1 and exemplify our considerations for assessing the property fulfillment hereinafter. For each approach, we provide relative ratings of each property.

Rating Methodology. As part of our assessment of related work, we rate data privacy (**P2**) based on the provided encryption capabilities as well as the level of control that the data owner maintains over the respective data and we now provide further details regarding our methodology of rating these qualitative aspects. Since Westerkamp et al. [29] propose unencrypted product information on a public ledger, we rate the respective data privacy level as ○. In contrast to this approach, ProductChain [38] encrypts production data, while all tracking and tracing references remain unencrypted, such that we assess its data privacy as ◑. Our own approach [40] further allows the encryption of tracking and tracing information, but the data owner cedes the control over the encrypted data to another party, such that we rate the data privacy level as ◕. Finally, MediLedger [79] allows data encryption, and the data owner is in sole control over her information, such that we rate the data privacy as ●. However, the requirement for the data owner to grant access to her data on demand negatively affects MediLedger's accountability ◑, which we usually rate as ● for blockchain-backed approaches. Similarly, we rate the multi-hop capabilities (**P4**) and product modeling (**P5**) based on the approaches' considered scope along the supply chain and their granularity regarding individual products and trades. Namely, we distinguish approaches that only consider single-hop protocol flows (○), multi-hop flows that partially cover the supply chain (◑), and approaches conceptually capable of handling complete supply chains (●). We follow a comparable approach when rating the remaining properties and explicitly state the respective utilized blockchain type if applicable.

Although specialized approaches exist for each property, a holistic (use case-independent) approach is still missing. While considerations of data privacy are treated only peripherally by many approaches (enhanced data privacy could

Table 1: Survey of related work; categorized by target application areas and sorted by year and name. We represent the fulfillment of a property by ●, partial fulfillment granularly by ◕, ◑ and ◔, and no fulfillment by ○. ? indicates that we are unable to assess the fulfillment reliably and - refers to properties that are not applicable to the respective work. Approaches that do not fully address product modeling are usually use case-specific, i.e., their properties cannot be generalized easily. While many specialized approaches exist (cf. properties), a holistic, general approach is missing.

Category	Publication \ Property	No-Trust Assumption (P1)	Data Privacy (P2)	Accountability (P3)	Multi-Hop Capabilities (P4)	Product Modeling (P5)	Scalability (P6)	Evaluation (P8)	Implementation* (P7)	Universality (P9)	Blockchain Type (P10)
Universal	Abeyratne and Monfared (2016) [19]	◕	○	●	●	●	?	○	○ / ○	●	Unspecified
	Peer Ledger (2016) [72]	◑	◔	●	●	●	◑	?	● / ●	●	MIMOSI
	Weber et al. (2016) [91]	●	◑	●	-	○	-	●	● / ○	●	Ethereum
	OriginTrail (2017) [89]	◑	◑	●	●	◔	◕	?	●[1] / ●	●	Generic
	Wu et al. (2017) [26]	◑	◔	●	○	◑	◑	◑	● / ○	◑	Unpecified
	Amborsus (2018) [73]	◔	○	●	◑	◔	◕	◔	●[2] / ●	●	Ethereum
	Gao et al. (2018) [71]	◕	◑	●	○	○	◔	●	● / ○	?	Hyperledger
	Kim and Laskowski (2018) [21]	●	○	●	●	◑	○	○	●[3] / ○	?	Ethereum
	Wang et al. (2019) [20]	◑	○	●	●	◑	◔	●	● / ○	●	Ethereum
	Pennekamp et al.♦ (2020) [40]	●	◕	●	●	●	◕	●	● / ○	●	Generic
Anti Counterfeit	Project Provenance (2013) [63]	○	◑	◑	●	◑	?	○	● / ●	○	?
	Everledger (2015) [17]	○	◑	◑	●	◑	?	○	● / ●	○	Hyperledger
	MediLedger (2017) [79]	◑	●	◑	○	●	?	○	● / ●	○	Ethereum
	Modum (2017) [80]	◑	○	●	○	◑	◑	◑	● / ●	○	Ethereum
	PharmaTrace (2017) [81]	◑	◑	●	◑	◑	?	○	◑ / ○	○	Hyperledger
	Toyoda et al. (2017) [22]	◑	○	●	◔	◑	◑	◑	● / ○	●	Ethereum
	FarmaTrust (2018) [27]	◑	◔	●	○	◑	◕	○	● / ●	○	Quorum
	Guardtime (2019) [82]	○	◑	●	●	◕	◑	○	● / ●	○	Unspecified
	Tahir and Hussein (2020) [84]	◔	○	●	◕	◑	◔	○	● / ○	○	Unspecified
Food Chains	Kelepouris et al. (2007) [64]	○	○	○	●	●	●	○	● / -	○	-
	Palaniswamy et al. (2008) [68]	○	○	○	◔	○	◕	●	● / -	○	-
	Pang et al. (2010) [65]	○	○	○	◑	○	●	●	● / ○	○	-
	Bahrudin et al. (2011) [67]	○	○	○	◕	●	-	○	○ / -	○	-
	Tian (2016) [23]	◑	○	●	◑	◑	○	○	? / ?	○	Unspecified
	IBM (2017) [18]	◑	◔	●	◑	●	?	◑	● / ●	○	Hyperledger
	Tian (2017) [66]	◑	○	●	◑	◑	◑	○	? / ?	○	BigchainDB
	Pincheira Caro et al. (2018) [24]	◑	○	●	●	●	◕	●	● / ○	○	Ethereum
	ProductChain (2018) [38]	◔	◑	●	●	●	◑	●	● / ○	○	Hyperledger
	AQUACHAIN (2019) [74]	◔	◔	●	●	◕	◔	●	● / ○	○	Generic
	Hyper-FTT (2019) [76] / FSCTS (2020) [93]	◑	○	●	●	◕	◔	●	● / ○	○	Hyperledger
	BMLFTS (2021) [78]	◑	○	◕	●	◑	◔	●	● / ○	○	Unspecified
Logistics	OpenPort (2015) [85]	◑	◔	●	●	◑	◑	○	● / ●	●	Unspecified
	SmartLog (2018) [86]	◔	◑	●	○	◔	◕	◑	●[4] / ○	●	Hyperledger
	Helo and Hao (2019) [28]	◔	◔	●	○	◔	◔	○	● / ○	●	Ethereum
	Shipchain (2019) [87]	◑	◑	●	○	○	◕	○	●[5] / ●	●	Ethereum
	Li et al. (2020) [88]	◑	◑	●	○	◕	?	○	? / ?	●	Unspecified
Industry	Goodrum et al. (2006) [69]	○	○	○	○	◑	●	●	● / -	○	-
	Nayak et al. (2015) [25]	○	○	○	○	◑	◕	○	○ / -	○	-
	Appelhanz et al. (2016) [62]	○	○	○	●	●	◕	◑	● / ○	?	-
	Figorilli et al. (2018) [30]	◑	○	●	●	◕	◔	◔	● / ○	○	Ethereum
	Westerkamp et al. (2018) [29]	◕	○	●	●	●	○	◑	● / ○	?	Ethereum
	Altmann et al. (2020) [94]	◔	◔	○	●	●	●	◔	? / ?	●	-

* □ / □: Implementation available / Implementation deployed (in Production) ♦ Initial publication of PrivAccIChain
[1] https://github.com/OriginTrail/ot-node [2] https://github.com/ambrosus [3] https://github.com/professormarek/traceability
[4] https://projectsmartlog.gitlab.io/smartlog-installer/ [5] https://github.com/ShipChain

be achieved without substantially changing the respective architectures), multi-hop capabilities and the achievable scalability are tied more closely to the considered design decisions. While centralized approaches promise extensive scalability, the scalability of blockchain-backed approaches significantly depends on the considered scenario (i.e., its universality), the granularity of product modeling, and the amount of handled, stored, and exchanged information.

The lack of universal approaches that provide an accountable multi-hop production model with data privacy considerations and match real-world supply chain scaling demands emphasizes the necessity for further research on collaboration in digitized supply chains. In the subsequent section, we further reason on the presented insights of related work to derive and formulate six design goals for architectures that address multi-hop collaboration in supply chains.

4. Design Goals for Private Multi-Hop Accountability in Supply Chains

Based on our literature review (cf. Section 3) and the desirable properties (cf. Section 2.4), we now derive a minimal set of design goals when providing *multi-hop accountability* for supply chains. This set of goals serves as a benchmark for current and future approaches, especially our proposed architecture, which we introduce in Section 5. Overall, we identify the six design goals of *accountability*, *verifiability*, *privacy preservation*, *security*, *scalability*, and *autonomy*. In the following, we present each goal as well as their interplay to create a common understanding for future use.

G1: Accountability. With respect to the desirability of accountability (**P3**), we explicitly demand that all involved parties need to be held responsible in case of identified misconduct, even in the absence of deliberately trusted parties (**P1**). Besides creating an immutable event log, processed and stored information regarding business operations, production processes, and product flows must be recorded persistently in a tamperproof manner. The recording of supply chain data must especially retain the ability for multi-hop tracking and tracing (**P4**) along supply chains. Hence, tamperproof and reliable data retention and provision of data to authorized parties are necessary [95]. Further, persistent guarantees on data existence, nature and ownership are required for investigating historical data in cases of alleged misconduct and for holding the misbehaving party responsible. Thus, accountability is crucial in industrial settings.

G2: Verifiability. As a precondition to achieve accountability (**G1**), recorded information must be verifiable retrospectively by all involved parties to guarantee that recorded data is available and untampered upon later investigation. While verifying data upon insertion could ensure its correctness, corresponding solutions would require a trusted verifier or global consensus, which would impede either our requirement to avoid especially trusted parties (**P1**) or the system's achievable scalability (**P6**). Hence, on-demand verifiability by all involved participants promises a more sustainable oversight to ultimately enforce accountability (**P3**), i.e., by distributing such capabilities among all parties.

G3: Privacy Preservation. While concrete supply chains may vary in their degree of required privacy protection, an infrastructure supporting multi-hop tracking and tracing must be able to properly protect data that is deemed sensitive, and restrict access to that data accordingly. Overall, businesses are especially cautious when sharing in-company and business-relevant data with other stakeholders. Privacy preservation (**P2**) thus constitutes another central design goal. However, it conceptually opposes the accountability (**G1**) and verifiability (**G2**) goals as the correspondingly required data might be sensitive and, thus, should be kept private in theory. Here, respective approaches must present individual solutions on how to balance this trade-off, i.e., to encourage participation while offering added value alike.

G4: Security. In accordance with data privacy (**G3**), we assess data security as equally important. In particular, businesses need to be able to rely on the enforcement of the demanded data privacy regulations regarding both malicious insiders and external attackers. Hence, any suitable architecture must provide appropriate security features that prevent unintended data extractions or manipulation, such as encryption, access control, and digital signatures. While simple designs without situational features might reduce the attack surface that must be considered for conducted security analyses, solutions that provide the demanded privacy, accountability, and verifiability features might require advanced designs with a dedicated focus on security. Hence, the aspect of security must be carefully weighted in all systems.

G5: Scalability. Infrastructures facilitating the global-scale management of supply chains must be able to operate accordingly. Therefore, we require that such infrastructures are able to represent, track, and trace a vast amount of products, even when large numbers of stakeholders are relevant, to satisfy the postulated scalability property (**P6**). Ideally, the system's performance is independent of the number of participants to allow for virtually unlimited and arbitrarily complex supply chains. However, scalability is not limited to computational power. It also refers to the imposed storage requirements, i.e., they must remain feasible and affordable in real-world scenarios as well.

G6: Autonomy. An infrastructure for the holistic management of supply chains must remain autonomous, i.e., manual interaction should only occur following exceptional events, such as claims regarding malicious behavior or data manipulation. Only by reducing the necessity for manual interactions, we can ensure reasonable system scalability (**G5**). The requirement for autonomy especially holds for our desired multi-hop features (**P4**), i.e., both multi-hop tracking and tracing must be operable without involving manual interaction with the affected companies. Similarly, the system should not depend on individual participants' availability as their presence cannot be guaranteed in volatile (and developing) business landscapes. To conclude, widespread automation is probably preferable in real-world settings.

Considering the potential lack of trust between multi-hop business partners (**P1**), we demand the retention of autonomy regarding the retrieval of already provided information, even in the presence of businesses that decide to revoke their cooperation regarding multi-hop collaboration. Thus, we require persistence and reliability of information by defining a form of non-repudiation for data-providing companies, i.e., we consider any published data as available.

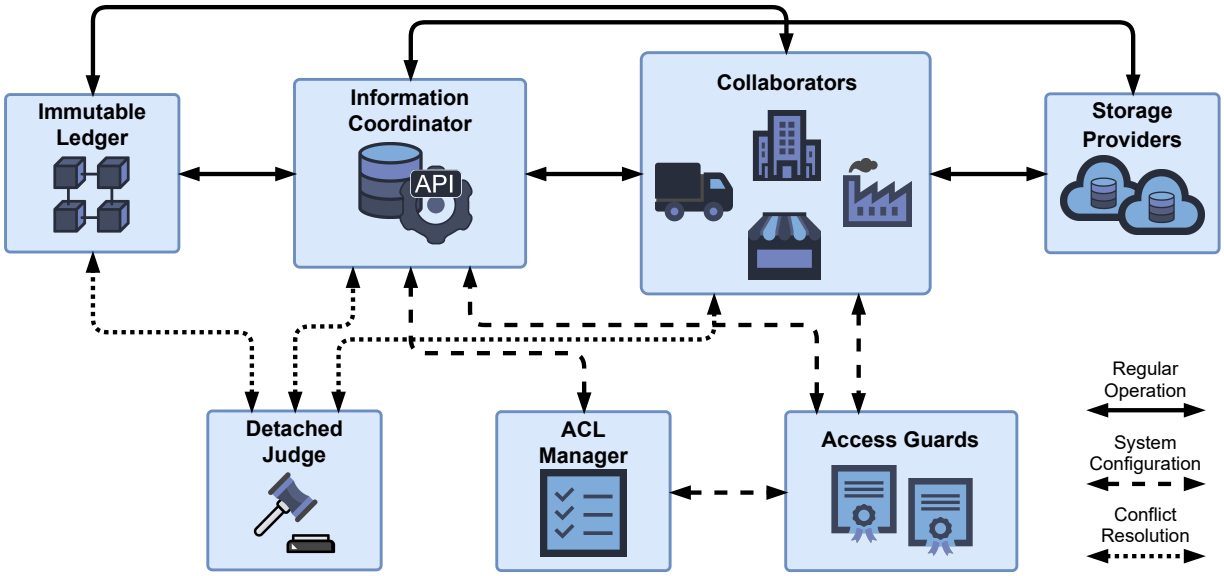


Figure 2: Overview of the architecture design with all involved entities and their respective relations. First, this illustration covers interactions during regular operation, i.e., the exchange of information that is necessary for multi-hop sharing of information on products and production. Second, dashed lines indicate the exchange of information for system setup and adjustments due to joining or departing participants. Third, the dotted connections with the detached judge represent potential interactions during conflict resolution.

Fulfilling these six design goals is sufficient to create a trustworthy and scalable infrastructure to facilitate multi-hop dataflows along supply chains, even in the presence of mutually distrusting stakeholders. The remaining properties which we presented in Section 2.4 and Section 3.3, while constituting important guidance for ongoing research in this domain, do not directly imply further design goals. Namely, we consider management infrastructures tailored to special use cases, e.g., incorporating distinct demands from the supply chains' product domain, viable if they fulfill our design goals even if they willfully neglect universality (**P9**) or fully specified product modeling (**P5**). Furthermore, depending on the concrete trust model, such infrastructures might be operable without a blockchain-backed storage. Hence, we do not consider utilizing blockchain technology (**P10**) mandatory for all solutions. Finally, while a proper evaluation (**P8**), supported by a publicly-available implementation suited for real-world use cases (**P7**), is required for the assessment of proposed solutions, they do not constitute explicit and initial design goals.

Based on the derived design goals, we now present our proposed architecture [40] for the accountable multi-hop collaboration along supply chains based on a blockchain-backed storage in detail. Afterward, we validate its conformance with our identified design goals in the subsequent sections of this paper.

5. A Blockchain-Backed Architecture Supporting Lightweight Multi-Hop Accountability

We now present *PrivAccIChain* (privacy chain), our **Priv**acy-preserving and **Acc**ountable multi-hop **I**nformation-sharing platform for supply **Chain**s, based on the design goals we identified in Section 4. First, we provide a design overview in Section 5.1 and introduce the entities involved in our design in Section 5.2. Afterward, we detail how PrivAccIChain protects dataflows based on attributed-based encryption (ABE) in Section 5.3 and provide guidelines for data access policies in Section 5.4. Finally, in Section 5.6, we present how information flows in PrivAccIChain to achieve a secure and trustworthy ledger for facilitating the multi-hop tracking and tracing of products, and we detail how PrivAccIChain utilizes fingerprints of provided data records to achieve reliable accountability and verifiability.

5.1. Design Overview

Our design of PrivAccIChain realizes the verifiable and thus accountable, yet by default, privacy-preserving information exchange for supply chains through a consequent separation of concerns. Our reasoning behind PrivAccIChain's design is that both accountability and privacy preservation protect participants from harm, be it to pinpoint

the origin of harmful incidents or the prevention of unintended leakage of company-internal information, e.g., to colluding, malicious entities. To this end, we rely on attribute-based encryption (ABE) [96–99], an encryption scheme that allows for many-to-many public key encryption, as well as blockchain technology to utilize an immutable, yet distributed log of all interactions [100]. To allow for reasonable flexibility, our architecture is tailored to supply chains with information flows that can be modeled as a DAG as we introduced in Section 2.

Figure 2 provides an overview of PrivAccIChain's architecture by showing the interplay of all involved entities. All data records relevant for the multi-hop information-based collaboration along supply chains originate at the *collaborators*, which represent individual companies along the supply chain. Our design acknowledges that excessive point-to-point communication between involved collaborators would be detrimental to the infrastructure's scalability (**G5**), and this approach would make collaborators increasingly dependent on each other (**G6**). To overcome both issues, we propose to relieve collaborators from the need for handling information flows and long-term storage themselves by securely outsourcing all data to a logically centralized *information coordinator*. The information coordinator only holds encrypted data and acts as a single point of contact, which handles all data provision and retrieval requests by collaborators. To further improve the scalability, collaborators can optionally share further use case-specific data, e.g., manufacturing data, with other entities by outsourcing this data to external *storage providers*, such as a dedicated cloud. If collaborators opt to use the service of an additional storage provider, the information coordinator only manages signed meta information about the outsourced data, while requests can still be answered centrally, although the information coordinator does not directly store all encrypted data records.

PrivAccIChain ensures the privacy preservation of each company's data records (**G3**) as well as their security (**G4**) through a decentralized management of ABE keys. We achieve this decentralization again by enforcing a separation of concerns. We enforce strict access control through ABE by allowing multiple independent *access guards* to issue individual attribute sets and decryption keys. Collaborators obtain this keying material based on an access control list (ACL) that centrally managed by an *ACL manager*. This way, the ACL manager has no access to the keying material required to satisfy any ABE policies. Furthermore, a subset of PrivAccIChain participants, and optionally external contractors, jointly maintain an *immutable ledger* that stores cryptographic fingerprints of data records to facilitate their verifiability (**G2**) and thereby provide (long-term) accountability (**G1**). Finally, a *detached judge* augments our design with an entity for dispute resolution, especially for exceptional cases that require manual investigation.

Based on these entities, the regular operation of PrivAccIChain is as follows. Collaborators submit their appropriately encrypted interactions along the supply chain to the information coordinator, who keeps the respective data records to respond to subsequent queries by other collaborators. The collaborators additionally provide cryptographic fingerprints of each data record uploaded to the information coordinator along with cryptographic signatures to the information coordinator, who stores each fingerprint as a proof of the existence and integrity of the submitted data record on the immutable ledger. Hence, the immutable ledger enables long-term verifiability and accountability in our system and serves as a reliable source of information for investigations in case of conflicts.

In conclusion, our combination of a logically centralized information coordinator enables the desired multi-hop information flows, while our decentralized application of ABE in conjunction with an immutable ledger maintains the privacy preservation, security, and accountability of all data records. In the following, we describe the introduced entities in more detail before highlighting different design aspects of our architecture.

5.2. Entity Introduction

To provide an in-depth understanding of the responsibilities of the entities in our architecture (cf. Figure 2), we now elaborate on their roles. We further highlight their respective contribution to fulfilling our design goals.

Collaborators. All users of PrivAccIChain, i.e., all participating companies, constitute collaborators. Each collaborator provides encrypted data records and may query information on products, sales, and production processes across multiple hops within the supply chain based on other collaborators' data records. For those operations, each collaborator interacts only directly with the information coordinator to avoid costly point-to-point communication among all collaborators along the same supply chain. Using this approach, we unburden collaborators from storing data records locally and ensure the availability of data records even if individual collaborators are untrusted (**P1**). Furthermore, this consequent outsourcing of data records mitigates the impact of malicious collaborators as they cannot provide incorrect data on-demand, e.g., when they are informed about being investigated. To cope with security (**G4**) and privacy (**G3**) concerns stemming from outsourcing data records to the information coordinator, PrivAccIChain enables

collaborators to specify fine-granular access policies for their data records using our ABE scheme (cf. Section 5.3). Our architecture protects collaborators further from dishonest collaborators by ensuring the verifiability (**G2**) of data records and thus achieving the accountability (**G1**) of malicious collaborators.

Information Coordinator. As a conceptually centralized entity, the information coordinator handles all data records submitted by the collaborators. It persistently stores these records, maintains a modification history, and grants data access to collaborators if they satisfy the requested records' access policies. We rely on this entity to support the transparency needed for multi-hop accountability (**P4**). In particular, the information coordinator provides an interface for three different types of queries, which require a previous authentication by the collaborators, i.e., (i) provisioning information, (ii) updating of existing data records, and (iii) requesting data records from other collaborators.

First, *information provisioning* allows collaborators to provide (encrypted) data records regarding products and production processes. During the information provisioning, the information coordinator and the data-providing collaborator agree on deterministic fingerprints to link to the respective data records before the information coordinator uploads these fingerprints to the immutable ledger. Second, the information coordinator enables the *updating of existing data records*. To maintain accountability (**G1**) and verifiability (**G2**), the coordinator allows replacing data records, but still retains the full version history of updated data records. Each update is addressed by a distinct fingerprint to disambiguate revisions of the same data record. Here, the information coordinator only stores incremental updates to remain scalable (**G5**) despite keeping this long-term record of all data. Third, collaborators need the possibility to *request specific data records*. Thus, the information coordinator provides functionality for the retrieval of one or multiple data records. Hereby, the information coordinator applies the access policy specified by the data-providing collaborator as the first mechanism for privacy preservation (**G3**). Since data records are encrypted, information leakage or a misbehaving information coordinator do not impede the confidentiality of the companies' data records.

We encourage to distribute the information coordinator across an independent consortium, which is unrelated to the other collaborators, to prevent immediate conflicts of interests and to mitigate a single point of failure. To this end, the information coordinator is only logically centralized, but can still operate in a distributed manner.

ACL Manager. Due to our application of ABE, collaborators must obtain attributes determining their privileges to access data records from the information coordinator. PrivAccIChain thus maintains an access control list (ACL), which keeps track of all valid attributes and which collaborators are assigned which attributes. Our design designates a special *ACL manager* to perform this task of assigning attributes to collaborators. This way, the ACL manager contributes to ensuring the security (**G4**) and privacy preservation (**G3**) of our architecture. However, to maintain mutual oversight (**P1**), the ACL manager does neither provide the keying material for the specified attributes, nor does it enforce the policies defined by collaborators. These tasks are performed by the set of independent access guards and the information coordinator, respectively. Again, we propose to further distribute the ACL manager's responsibilities among multiple stakeholders, e.g., involving external stakeholders such as governmental agencies or associations.

Access Guards. As stated before, we prevent an excessive concentration of authority at the ACL manager by ensuring that that entity cannot enforce arbitrary access policies. In addition to the information coordinator rejecting unauthorized requests for data records, a set of independent *access guards* maintains individual sets of attributes as well as their associated keying material. Particularly, each access guard provides eligible collaborators with the attribute-private keys required for record decryption to those collaborators that have been assigned the respective attributes by the ACL manager. Hence, granting access to data records is distributed among at least four entities, i.e., the information coordinator, the ACL manager, and multiple access guards, reducing the risks of illegitimate data access (data security (**G4**) & privacy preservation (**G3**)). Due to a partitioning of the attribute space, a single access guard cannot control all attributes at once and thus has no decryption capabilities by itself, i.e., no specific trust is needed (**P1**). We further elaborate on the accompanying policy design process considering multiple, independent access guards in Section 5.4.

Immutable Ledger. The information coordinator accumulates the encrypted data records of all collaborators in a centralized manner, and is responsible for ensuring the availability of all data records. To improve the verifiability (**G2**) and accountability (**G1**) of this data handling, PrivAccIChain requires to store cryptographic fingerprints of all data records and operations thereon on an *immutable ledger* that is based on a consortium blockchain [101]. Thereby, the immutable ledger constitutes an append-only log of all available data records and related events. Since the immutable ledger only stores fingerprints of data records and no production data, the ledger does not introduce any privacy issues. Even though the information coordinator is the main writer to this ledger, the ledger itself is maintained by an independent consortium of other stakeholders consisting of, e.g., collaborators and auditing third parties. Relying on a single entity to issue transactions mitigates potential issues regarding the anonymity of the collaborators,

i.e., production patterns are relayed and not visible by third parties. The security of the cryptographic fingerprints, which we further detail in Section 5.6, and the independent consortium-internal oversight make any misconduct by the information coordinator immediately apparent to the affected collaborators (enabling verifiability (**G2**)). Thereby, the information coordinator can be held accountable (**G1**). Furthermore, investigated collaborators now cannot collude with the information coordinator anymore after the fact, e.g., in attempts to modify data records that could potentially unveil wrongdoing by the collaborator.

Storage Providers. We allow collaborators to augment our information coordinator-centered design by additionally storing use case-specific data with third-party *storage providers*, e.g., cloud storage providers. In this case, the collaborators only share a reduced data record with the information coordinator, which provides the storage location of the outsourced data as well as any relevant metadata and decryption material. Hence, we massively increase the flexibility of our system by allowing collaborators to store data either with the information coordinator directly or, possibly in private (privacy preservation (**G3**)), with third-party storage providers. Furthermore, external storage providers unburden the information coordinator from having to store all data and thereby increase PrivAccIChain's overall scalability (**G5**). Finally, a federated storage distributed among multiple providers increases the flexibility regarding use case-specific requirements, e.g., supporting custom payloads and data formats (universality (**P9**)).

However, data outsourced to an external storage provider simultaneously constitutes a loss of control for the information coordinator and thereby has the potential to impede the overall autonomy (**G6**) of PrivAccIChain and the verifiability (**G2**) of the data it maintains as service providers can delete or withhold such outsourced data. To cope with these newly introduced risks, we explicitly hold data-outsourcing collaborators accountable for data loss originating from a third party. Pinpointing a fraudulent storage provider is still possible in this approach as the information coordinator, and the immutable ledger still hold all metadata required to verify a record's existence and integrity.

Detached Judge. Finally, our design considers a *detached judge* for on-demand dispute resolution. While our architecture provides accountability (**G1**) and verifiability (**G2**) as well as well-defined behavior boundaries for all entities, we do not automate punishing misbehaving entities, e.g., via financial penalties. This design decision acknowledges the high complexity of dispute resolution in real-world settings. Disputes may, for instance, arise regarding the value or correctness of data records, or collaborators may dispute the decisions of the ACL manager, the access guards, or the information coordinator. Since these entities are designed to mutually control each other, a lack of consent on ACL manager decisions or access control implementations that do not follow the ACL manager's decisions are conceivable. However, determining which entity misbehaves is a complex and non-trivial decision as it depends on underlying contracts that are not necessarily part of our proposed architecture. To resolve this tension, we thus allow for manual dispute resolution by the detached judge, who has to be considered trustworthy by all participants. This approach to conflict resolution is in line with related blockchain-backed platforms that need to handle disputes regarding on-chain data records [102, 103]. Hereby, we further contribute to the fulfillment of accountability (**G1**), while the possibility for independent external verification (and auditing) also increases the trust in our architecture.

Following our in-depth description of all entities, we now provide further insights into our proposed hybrid encryption scheme, which is in place to ensure data confidentiality and to prevent illegitimate access to sensitive information.

5.3. Hybrid Encryption Scheme

Data privacy (**G3**) and security (**G4**) are key aspects of our architecture, which we achieve by encrypting all data provided by the collaborators before handing it over to the information coordinator. In this section, we present our hybrid encryption scheme that PrivAccIChain uses to ensure data confidentiality and to provide fine-granular, policy-based access control to collaborators. Our encryption scheme uses a hybrid approach based on symmetric payload encryption as well as ciphertext-policy attribute-based encryption (CP-ABE) [98, 99, 104] for distributing the symmetric payload keys. In Figure 3, we visualize the concept of our hybrid encryption scheme.

Encryption Process. Before releasing her data records to the information coordinator, the collaborator encrypts each data record and potential tracking and tracing references using an ephemeral symmetric payload key. Afterward, the collaborator specifies the access policies for each data record as an ABE-compatible Boolean formula and encrypts the symmetric payload key accordingly using CP-ABE. Finally, the collaborator uploads both the symmetrically-encrypted record and the CP-ABE-encrypted symmetric payload key to the information coordinator. As the encryption can be applied gradually, our scheme supports that different access policies are defined for subsets of the payload.

Decryption Process. Other collaborators can now obtain data records from the information coordinator as long as the collaborator's assigned attributes are compatible with the data records' access policies. In this case, a collaborator

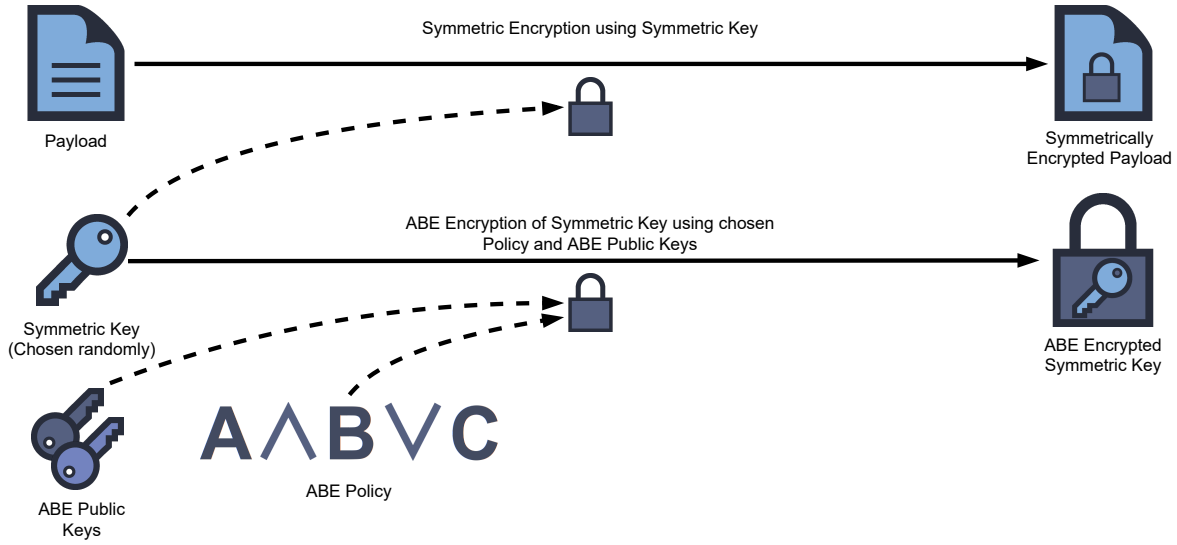


Figure 3: Our proposed hybrid encryption scheme relies on symmetric encryption for encrypting the payload, while we utilize attribute-based encryption (ABE) to encrypt the symmetric key used for the payload encryption. The resulting encrypted key is then distributed to the desired collaborators.

can use the corresponding attribute private keys and decrypt the symmetric payload key according to the access policy using CP-ABE. Finally, the collaborator obtains the plain data by decrypting it using the symmetric payload key.

Design Implications. Our hybrid encryption scheme is designed to combine the preferable performance of symmetric encryption schemes with the fine-granular access control of CP-ABE. In contrast to traditional asymmetric encryption schemes, the complexity of ABE schemes depends on the complexity of the desired access policy rather than the number of potential recipients. Hence, CP-ABE allows the bundling of larger groups of collaborators using only a few attributes to maintain PrivAccIChain's scalability (**G5**) even if single supply chains engulf many collaborators eligible to perform multi-hop queries. Second, representing groups of collaborators with attributes mitigates the risk of discriminating individual collaborators during the key distribution process (cf. Section 8.2). Since all collaborators receive the same encrypted symmetric key, validation of its correctness is possible by each collaborator with access to the record. If the key is invalid, this (un)intentional issue can trigger the conflict resolution.

Overall, our hybrid encryption scheme is well-suited for large and dynamic environments as all underlying computations are independent of the number of recipients. Consequentially, we achieve scalability while maintaining security. As the related data privacy is directly influenced by the used ABE policy, we next take a look at its design.

5.4. Policy Design

PrivAccIChain's hybrid encryption scheme relies on CP-ABE [99] to enable collaborators to fine-granularly express their desired access policies. In this section, we discuss how collaborators can choose these Boolean policy formulas, consisting of conjunctions and disjunctions, to ensure favorable privacy preservation (**G3**) and data security (**G4**) while maintaining scalability (**G5**). The policy design mainly affects two properties, namely (i) the *collusion resistance*, i.e., the number of access guards that have to collude for illegitimate data record decryption capabilities, and (ii) the achievable *granularity* of the access control mechanisms. We evaluate the policy design in Section 7.3.1.

Collusion Resistance. Collaborators can improve the *collusion resistance* of their data records by requiring attributes issued by different access guards for decryption, i.e., by defining a conjunction of attributes. A policy $P = A \wedge B \wedge C$, where each attribute is issued by a different access guard, requires all these access guards to collude in contrast to the policy $P' = A$, where only a single malicious access guard can bypass the collaborator's policy. Thus, incorporating attributes from multiple access guards increases collusion resistance, but at the same time, such policies impede the performance of encryption and decryption operations (cf. Section 7.3.1). Collaborators can thus carefully trade off the security level of their data records against their required performance, and they can do so for each record. For instance, collaborators can protect highly confidential data records using attributes from all available access guards,

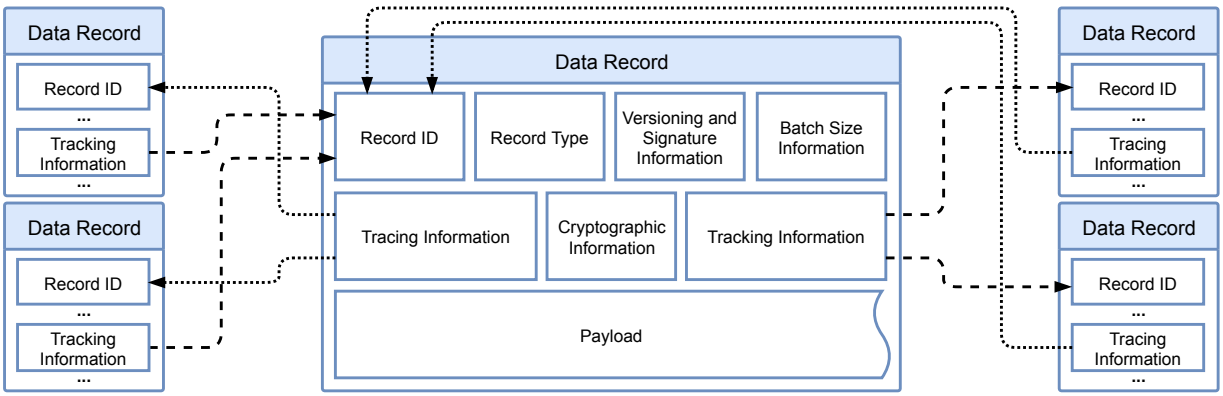


Figure 4: The record layout of our operations contains (encrypted) tracking and tracing fields that reference other records (double-linked record structure), corresponding to the structure of the DAG. Thereby, we enable collaborators to conduct multi-hop tracking and tracing along the supply chain. For encrypted information, the corresponding queries are only possible if the collaborator possesses the corresponding ABE attributes to successfully decrypt these fields.

while they may deliberately choose weaker collusion resistance for mass data they deem less confidential. Since the requirements for collusion resistance vary between different supply chains, we cannot provide a definite answer on the recommended conjunction length, but encourage a careful policy design based on these influencing aspects.

Granularity. By carefully considering the attributes' semantics, collaborators can furthermore tweak the *granularity* of their access policies. Each attribute identifies a set of collaborators or even individual collaborators. Collaborators can determine the best-suited attributes to use in their access policies by consulting the ACL manager. Hence, they can identify smaller sets of relevant collaborators per attribute for increasingly flexible access control. Appropriate use of attribute conjunctions and disjunctions allows for joining and intersecting these sets, which positively affects the achievable granularity and expressiveness of access policies. However, increasing the complexity of access policies, e.g., by creating long disjunctions of attributes corresponding to respectively small sets of eligible collaborators, may again negatively impact encryption and decryption performance (cf. Section 7.3.1).

Although appropriate attribute semantics depend on the targeted supply chain, we advise against assigning attributes to a single collaborator to avoid extensive performance losses as well as to mitigate the risk for discrimination of collaborators during the key distribution process (cf. Section 8.2). We further emphasize that the overall appropriate policy length is in the magnitude of the depth of the corresponding supply chain DAG (cf. Section 2.1) and not directly linked to the total number of collaborators in our system. Only attributes that are used for encryption and decryption affect the performance of our encryption scheme, i.e., the total number of available attributes has no impact.

Following the descriptions of the integrated building blocks of our architecture, we now take a look at the layout of our data records that enable all supported interactions and queries in our design.

5.5. Accountable Record Provision, Retrieval, and Updates

The collaborators share information with each other by storing data records at the information coordinator to enable different queries within the supply chain environment. In this section, we elaborate on the structure of PrivAccIChain's data records, which we visualize in Figure 4, as well as available types of data records and the interactions between collaborators and the information coordinator they are involved in.

Data Record Structure. Data records are organized as dictionaries of key-value pairs for maximum flexibility. Each data record has a unique `Record ID` and a well-defined `Record Type`. In PrivAccIChain, we distinguish between *produce records*, which represent production or assembly steps, final products, and services such as transportation, and *trade records*, which represent the transfer of ownership of a previously produced object from one collaborator to another. For better scalability (**G5**), we also support the registration of product *batches*, i.e., large amounts of identical products are represented as a single batch, with a `Batch Size Information` field denoting the number of products in the batch for tracking purposes. The `Record ID` should correspond to a unique mapping from physical goods to a digital identifier. Although we do not restrict the techniques of such a mapping, the utilization of

RFID tags, physical and digital certificates, or printed identifiers such as QR codes constitute appropriate options for different scenarios [105]. Furthermore, each data record contains versioning information, i.e., metadata that enables the reconstruction of every version of a data record, as well as signature information (`Versioning and Signature Information`). Collaborators sign a cryptographic fingerprint derived from the data record before the data record is sent to the information coordinator. This way, the information coordinator cannot tamper with the collaborator's data records without violating this signature [106].

The `Payload` field contains arbitrary data that is related to the respective product and its production steps. When relying on an external storage provider, the payload of a data record holds a reference to the location of the data at the storage provider as well as potential additional information. We store references to other data records in dedicated `Tracing Information` and `Tracking Information` fields. These fields basically contain (encrypted) pointers to another `Record ID` and indicate that the linked-to products contributed to the product, which the data record at hand references to. We distinguish between the payload and these references for improved automation as well as a reduced versioning overhead, since tracking references are usually inserted belatedly. This structure results in doubly-linked data records, where tracking and tracing operations correspond to following the respective references. Our data record format supports selective encryption, i.e., different parts of the payload and the references can be encrypted differently with individual policies for fine-granular access control. Information on encrypted data record sections as well as the corresponding key material are stored in a dedicated field called `Cryptographic Information`.

Record Types. Although we decided to utilize two types of data records, *trade records* can be omitted in certain scenarios, since all embedded information can also be included in the *produce record*. Scenarios with specific needs might require the differentiation between additional record types, e.g., a separation of production and transportation data. Our flexible data record format supports such modifications without excessive changes to our architecture. However, the resulting trade-off between simplicity and expressiveness must be considered here.

Interactions and Queries. The information coordinator provides an interface for the collaborators to provide, update, or retrieve data records. For updates and data retrieval, the information coordinator validates the access policy associated with the data record, which is stored as a part of the `Cryptographic Information` field. All data records are solely identified via their `Record ID`, which benefits the performance and reduces the indexing overhead at the information coordinator. For data record updates, the information coordinator only persists the changes with respect to the previous version to minimize the storage overhead. With the help of the maintained versioning information, each version can be reliably derived. Hence, the accountability and verifiability properties of our design are intact.

Given that the security guarantees of architecture depend on the verification of fingerprints of all processed data records, we next introduce the corresponding mechanism in more detail.

5.6. Blockchain-Based Accountability through Fingerprints

The collaborators rely on the information coordinator to handle the provided data records correctly. For data provision, a collaborator hands the control over the encrypted data over to the information coordinator. During subsequent data retrievals, both the data-providing and the data-retrieving collaborator need to be able to verify that they received the correct, untampered data record. Further, data-providing collaborators rely on the information coordinator to not delete any provided data records. Given the potential lack of trust between collaborators and the information coordinator (**P1**), we utilize the immutable ledger for retaining verifiable proofs of existence and originality of data records and enable collaborators to hold the information coordinator accountable for wrongdoing. The existence of these decentralized proofs allows for an autonomous (**G6**) verification of data records as well as an accountable identification of the responsible party in case of data unavailability or incorrect data records (**G1** & **G2**). Even though the information coordinator stores the collaborator's signature alongside each data record to prove the data record's integrity, PrivAccIChain relies on the immutable ledger to reliably, i.e., immutably, link the data records' IDs and fingerprints to their respective payloads to unveil, e.g., the absence of data records.

We utilize fingerprints of data records to achieve the desired accountability (**G1**) and verifiability (**G2**) features. We implement a deterministic algorithm to derive a fingerprint based on a data record. Consequentially, every entity that has access to the record can derive the fingerprint (also from encrypted payloads) and verify the record's originality. The possibility to derive the fingerprint without accessing the data record's plaintext is crucial to maintain data privacy (**G3**) and to enable the information coordinator to derive and verify fingerprints. The derivation of a fingerprint basically consists of calculating a cryptographic hash of a textual representation of the data record fields. Hereinafter,

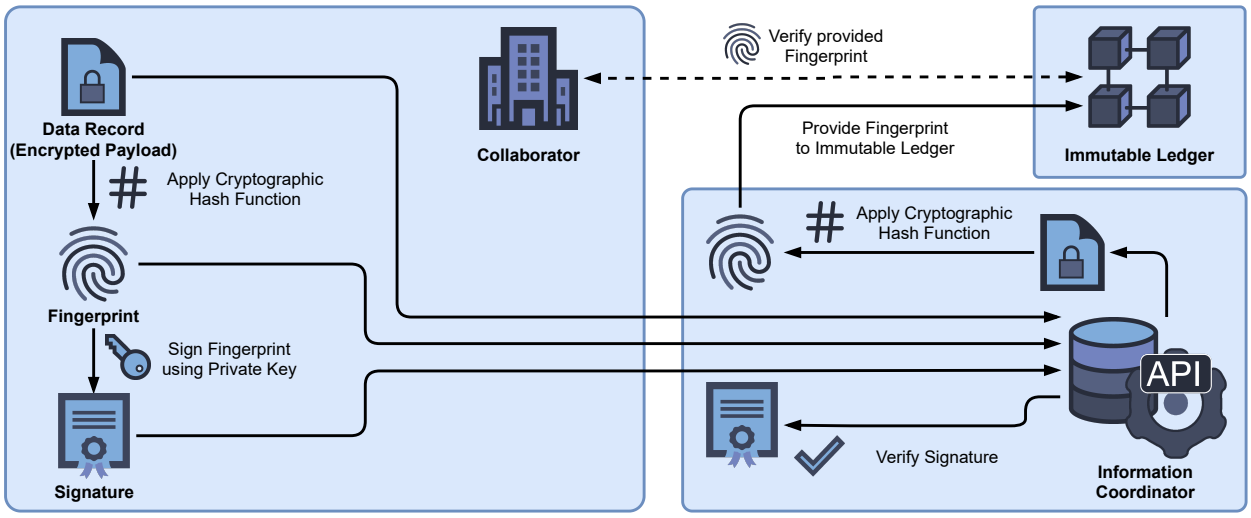


Figure 5: Collaborators derive a deterministic fingerprint from the encrypted data record. They provide the data record, the fingerprint, and a signature over the fingerprint to the information coordinator. The information coordinator verifies the correctness of the fingerprint as well as the signature and stores the fingerprint on the immutable ledger. Finally, the collaborator verifies that the correct (unmodified) fingerprint has been persisted on the ledger.

we describe the data record's integrity verification process for data record provision in Section 5.6.1 and during data record retrieval in Section 5.6.2.

5.6.1. Data Correctness Verification during Information Provision

During data record provision, the providing collaborator and the information coordinator agree on a deterministic fingerprint. After signing the fingerprint as well as the associated record ID, the information coordinator stores both values on the immutable ledger. In Figure 5, we visualize the derivation and verification. In this section, we now present the *fingerprint derivation*, how the information coordinator *commits* the fingerprint to the immutable ledger, how the data-providing collaborator *confirms* that fingerprint's correctness, and how we handle *updates* of data records.

Fingerprint Derivation. Before providing the encrypted data record to the information coordinator, the collaborator derives the respective fingerprint with the help of a cryptographic hash function. The collaborator signs this fingerprint with its private key to obtain a publicly verifiable signature. Afterward, the collaborator transmits the data record, its fingerprint, and the signature to the information coordinator, who verifies the signature and the fingerprint's correctness. In case of an incorrect fingerprint, the information coordinator rejects the current provisioning request. In cases where the fingerprint's correctness is disputed, both parties consult the detached judge for conflict resolution.

Committing the Fingerprint. Given a correct fingerprint, the information coordinator stores both, the data record's ID as well as the fingerprint on the immutable ledger. The associated transaction is signed by the information coordinator, such that the transaction serves as a PrivAccIChain-wide proof of existence and originality of the data record, similar to the approach presented by Zhao et al. [107]. The provision of the fingerprint to the immutable ledger is no time-critical operation. Hence, even delays of multiple hours or days are still acceptable. This tolerance enables optimizations such as aggregated transactions or meta fingerprints, which we further discuss in Section 7.3.5.

Fingerprint Confirmation. Once the fingerprint is stored on the immutable ledger, the data-providing collaborator verifies its correctness. If this check is successful, no further action is required from the data-providing collaborator. Thus, collaborators can *silently approve* the correctness of fingerprints to reduce the number of transactions to the immutable ledger. Otherwise, for incorrect fingerprints, the collaborator issues a transaction to the immutable ledger, which includes the claimed correct fingerprint. This objection is recorded persistently and triggers our dispute resolution process. While direct communication between the information coordinator and the collaborator should usually resolve fingerprint issues, involving the detached judge might be required. As the absence of an objection corresponds to the collaborator's agreement, we require objections to be entered within a well-defined period, e.g., within 48 h after the fingerprint has been stored on the immutable ledger by the information coordinator.

Record Updates. For data record updates, the fingerprint derivation and provision process is analogous. However, the information coordinator requires the collaborator to fulfill the access policy of the updated record. Furthermore, we slightly adapt the fingerprint derivation process, such that the fingerprint now includes the existing data record ID.

While the fingerprints stored on the immutable ledger ensure the absence of manipulation as well as the existence of a specific data record similar to the approach of Li et al. [106], they neither ensure the correctness nor the utility of the included information. In this context, dedicated solutions are needed [105]. Nevertheless, the fingerprints' existence ensures that the information coordinator can be held accountable for any manipulation or deletion of data, while the providing collaborator can be held responsible for invalid, manipulated, or made-up information.

5.6.2. Verification of Data Correctness during Information Retrieval

While the fingerprint verification during data record provision and updating provides guarantees to the providing collaborator, data-retrieving collaborators can later verify the record's integrity as well. Since the fingerprint and the providing collaborator's signature are stored along with the actual data record by the information coordinator, collaborators can utilize this information to verify the originality of the data record without interacting with the immutable ledger. This design improves the scalability (**G5**) and reduces the complexity of the verification process. Since all fingerprints are publicly available, the retrieving collaborator can verify the fingerprint autonomously (**G6**) without consulting the data record's original provider. If a data-receiving collaborator requests a data record and the information coordinator claims that the data record does not exist, the requesting collaborator can verify this claim based on the information persisted on the immutable ledger. As for the verification of originality, no interaction with the providing collaborator is required for existence verification. The verification process relies on the information coordinator to provide the requested data record to the collaborator and, in case of a fingerprint mismatch, the correct operation of the immutable ledger. We discuss the potential security implications of misbehaving or failing entities in Section 8.2.

Altogether, we achieve reliable and autonomous verification of originality and existence of data records for all authorized collaborators as well as the information coordinator. Our decision for silent approval reduces the number of transactions at the immutable ledger as well as the ledger's involvement in fingerprint matching verification (**G5**).

6. Insights into Car Manufacturing at e.GO Mobile AG

In the following, we rely on a real-world example to highlight the current situation and identified needs in industry. In particular, we look at an original equipment manufacturer in the automotive industry. In this context, we present an overview of the assembly line and the supply chain of the e.GO Mobile AG. The company was founded in 2015 as a manufacturer of electric vehicles and delivered its first vehicle model, the e.GO Life, to the customers in 2019. First, in Section 6.1, we describe the specific challenges that currently surface in the automotive industry due to the new advances such as the Internet of Production (cf. Section 1). These aspects mainly evolve around the management of interorganizational collaboration. Furthermore, we detail the expected benefits for e.GO and comparable car manufacturers resulting from tracking and tracing components in the supply chain on a global scale. Second, in Section 6.2, we explain why supply chain information sharing and corresponding accountability and transparency is a suitable source for competitive advantages based on the manufacturing of the e.GO Life.

6.1. Managing Interorganizational Collaboration in the Automotive Industry

Globalization and technological progress change the way how organizations act and the terms of managing the boundaries between organizations and their environment [108, 109]. The automotive industry, with its nowadays globalized supply chains, has been affected by this change for over two decades [110], while the business model behind the ecosystem is required to adapt due to disruptive technologies such as e-mobility, autonomous driving, connectivity, and digitization. To improve their productivity and to reduce costs, organizations within this industry and their environments should rapidly react to these dynamic and agile market situations [6, 111]. These new business models and the benefits of interorganizational collaboration urge organizations in the mobility market to adapt their structures accordingly. As a consequence, new forms of customer relationship management as well as innovative, data-driven business models can lead to a competitive advantage [112]. In this regard, e.GO utilizes a customer-focused agile product development that focuses on multiple stages of feedback to iterate the development of the electric vehicle while shortening the whole development time by using these agile and continuous feedback loops [113].

Establishing new Relationships. The described (current) evolution in managing the development process is challenged by the establishment of new, originally low-trust, business relationships with new suppliers or key partners at each stage of the iterative development. Thus, the improved accountability of (shared) information corresponding to supplied components ultimately improves the trust in these (new) suppliers as well. Consequentially, an architecture supporting the desired multi-hop information sharing (cf. Section 2.4) is highly beneficial, especially since an increased number of suppliers can be considered within each procurement step. Furthermore, as a newly founded company, e.GO's span of relationships with its suppliers, as well as its network of suppliers, is still limited compared to established original equipment manufacturers. However, these new dynamic relationships, enabled through multi-hop information sharing, are also relevant for established manufacturers as they face new powerful suppliers (e.g., CATL or LG Chem) and competitors (e.g., Tesla or e.GO) following the transition towards electric mobility [114].

Beyond that, manufacturers of electric vehicles are highly interested in maintaining their environmentally friendly impact, i.e., their direct suppliers as well as their complete supply chains must fit the respective footprint strategies. Taking this background into consideration, the benefits of PrivAccIChain are essential for the whole automotive industry as a transparent and accountable, yet business-friendly and privacy-respecting environment within the whole supply chain is needed to successfully manage the transition to deepened interorganizational collaborations.

Demand for Tracing. Apart from the previously mentioned accountability features in a low-trust, highly dynamic environment, PrivAccIChain can also contribute to realize several related business benefits. For e.GO, ensuring that all car components have a high level of traceability within the increasingly extending supply chain structures in the automotive industry is of utmost importance. The collective shift in the automotive industry, in terms of a more regional integration of component procurement, has led to a great dependency on these local firms and their respective supply chains in the emerging markets, such as Brazil, India, and China [115]. Hence, out of this perspective, e.GO is currently challenged by potentially opaque supply chain structures of their direct suppliers. Hence, PrivAccIChain enables tracing and reports of any errors or failures regarding software or hardware of the car directly to e.GO. As a consequence, the responsible supplier can be contacted, and both companies can react immediately to jointly resolve any issues. Appropriate actions can range from an update of production parameters over software updates to product recalls. To this end, PrivAccIChain's capability of tracing errors in more detail along the supply chain is combined with a very focused tracking of already sold products in the market to achieve sophisticated product information retrieval functionalities. Consequentially, this aspect is a key advantage of supply chain environments utilizing PrivAccIChain and an immense business benefit for the production of complex products, such as an electric vehicle.

Improved Processes. Beyond that, in relation to technology and innovation management topics, a second business benefit for e.GO is the supported general operation transparency. Raw material costs, profit margins in the supply chain, and the availability of innovation investment opportunities are crucial for manufacturing companies [116]. Hence, by mapping all supply chain and manufacturing dependencies, selectively benefiting from digitization-driven changes is an opportunity to obtain full transparency in terms of components and margins in the value chain. For example, dynamically reacting to customer change requests is significantly eased once all information has been registered in a digital way, i.e., the whole value chain is aware of the change request's impact. This value chain transparency enables future-oriented adjustments on the car, which themselves perfectly fit to the already used components.

Following this general overview of manufacturing in the automotive industry, we take a look at the manufacturing of a specific vehicle next, i.e., the e.GO Life.

6.2. Manufacturing of the e.GO Life

The e.GO Life is a typical urban city car that was developed for the low-cost small electric vehicle market. To achieve profitability, many changes to the traditional vehicle concept were essential. Especially the high costs of the electric battery required a critical breakdown of the supply chain and the manufacturing of the vehicle. With a colored-through thermoplastic exterior, no paint shop or press plant is required as part of the manufacturing process. As the exterior significantly affects the perceived quality of the vehicle, the preceding steps of the supply chain should guarantee high-quality products. As such, e.GO identified the necessity for accountability and an at least superficial exchange of production parameters to ensure the quality of the final product.

Manufacturing Steps. Additionally, the development and procurement strategy for the e.GO Life is focused on off-the-shelf fully assembled modules from traditional suppliers and other original equipment manufacturers to reduce the development costs and time in comparison to traditional vehicles. The mentioned strategy exposes the company

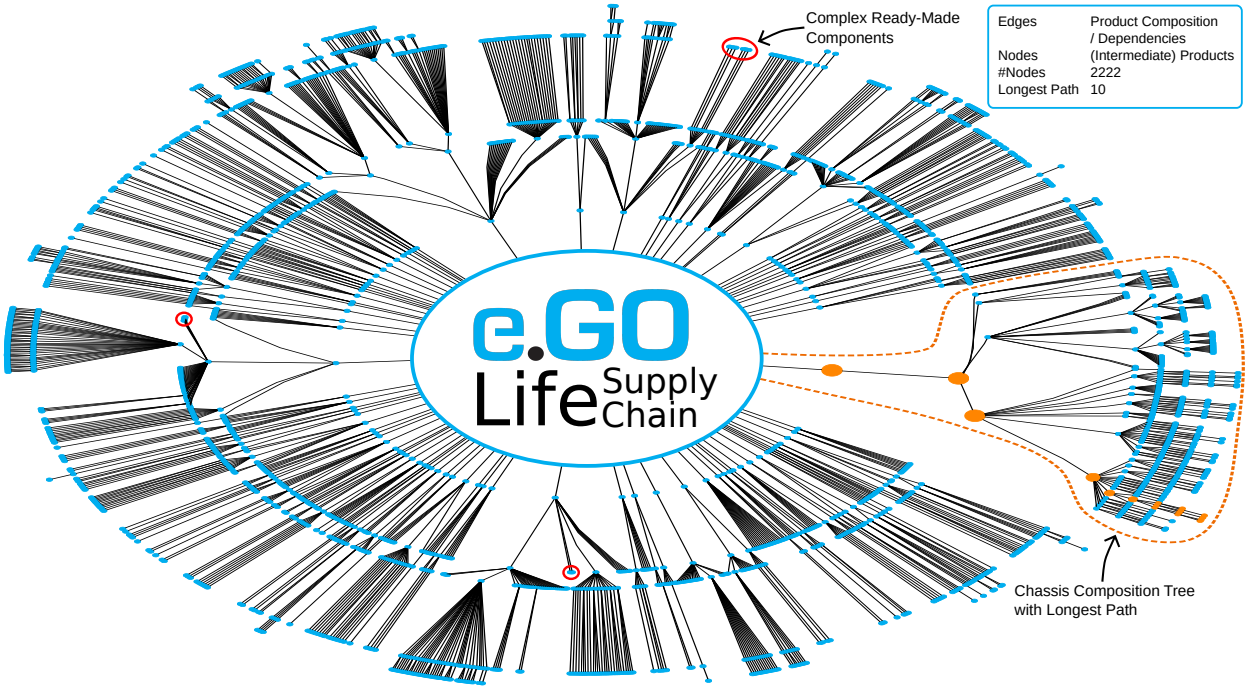


Figure 6: The supply chain of our chosen example, the e.GO Life, consists of ten levels: Nodes correspond to parts and intermediate products, while edges illustrate the dependencies of a production step, i.e., the product composition. While the center of the graph represents the assembled e.GO Life, the subtrees correspond to individual components. The granularity of the structure increases with the number of the selected level, i.e., nodes in the graph (subcomponents) depend on their children. We exemplary circle a subset of all utilized complex, ready-made products in red. The highlighted tree on the right corresponds to the assembly of the chassis of the e.GO Life. Due to its relevance for our evaluation, we further mark the longest paths (with depth 10) in the graph by coloring involved nodes in orange.

to the risk of lacking transparency in the supply chain due to the ready-made subcomponents and their dependencies. Hence, due to the provided accountability, PrivAccIChain is able to reduce this uncertainty as well as all associated risks to a minimum. To put the supply chain of an e.GO Life into perspective, we visualize its real-world structure in Figure 6. The visual structure of the graph highlights that the structure of an electric vehicle is complex, i.e., concerns several subcomponents, and consists of multiple levels. The complexity further reveals the large number of suppliers that e.GO is depending on. Here, the suppliers of each component and subcomponent could be switched dynamically depending on the current situation (e.g., supply bottlenecks or specific customer change requests). Every node, residing from level three up until nine, can be interpreted as a subcomponent or intermediate product. Especially on levels three and four, leaves (exemplarily circled in red) respond to ready-made products with hidden complexity. For example, these products can correspond to purchased electronic control units (ECUs), which themselves consist of a (complex) multi-staged supply chain. In contrast, levels eight to ten mostly consist of so-called c-parts, such as screws, nuts, or rivets. As another example, we highlight a higher node density on the right in orange, which represents the internal manufacturing of the e.GO Life's aluminum spaceframe and underlines the realism of the illustrated graph.

Keeping the complexity of an electric vehicle, exemplified by the e.GO Life, in mind, we provide a performance evaluation based on this real-world product in the next section. We estimate that these results can be transferred to other industries, given that automotive manufacturing (and its supply chain) is characterized by high complexity.

7. Evaluation of PrivAccIChain Based on a Real-World Supply Chain

To assess our architecture's suitability for real-world supply chains, we created a Python-based implementation for all relevant entities and analyzed its performance to evaluate the fulfillment of our scalability requirements (**G5**). First, we present our experimental setup and utilized technologies in Section 7.1 and the model derivation process in Section 7.2 corresponding to our real-world example (cf. Section 6). Afterward, we evaluate the performance of

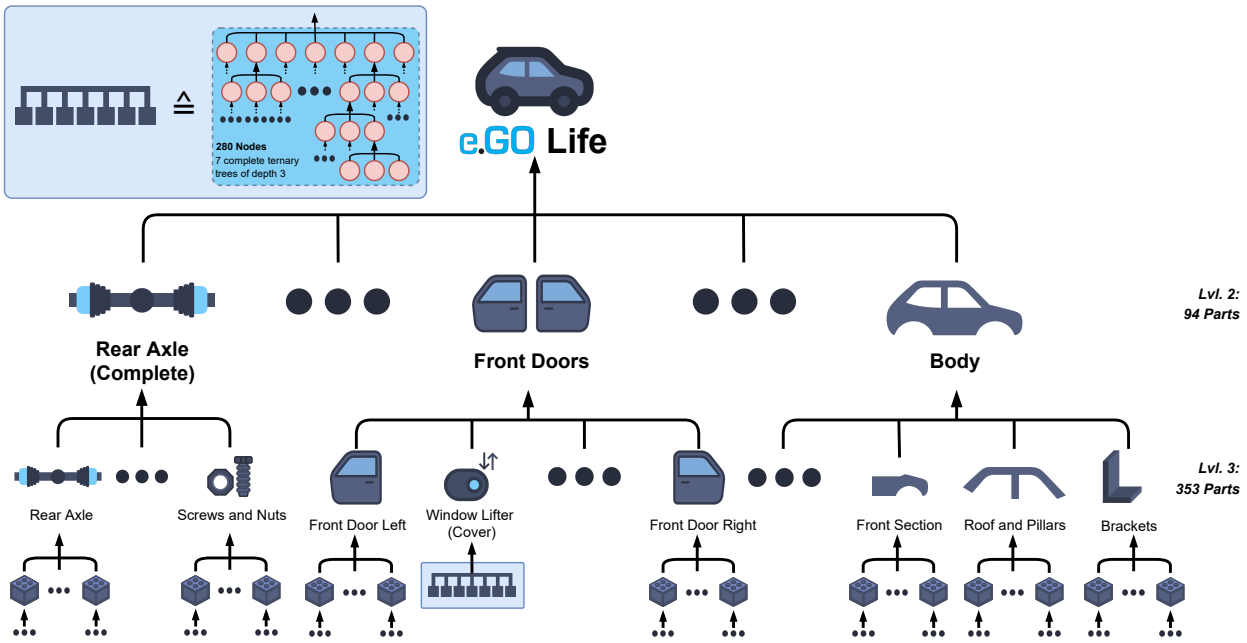


Figure 7: This simplified detailed visualization of a subset of the composition graph that we presented in Figure 6 shows the branching behavior for different production levels of an e.GO Life. The complete graph comprises more than 2000 nodes and spans 10 levels. We extend this *base model* with artificial production dependencies (on level three and four) to account for pre-assembled components in the original graph. We refer to this model as *extended model*.

our hybrid encryption scheme (Section 7.3.1) and the provision and update of data records (Section 7.3.2). We then analyze the retrieval of data records as well as the associated use cases for tracking and tracing based on a realistic automobile (Sections 7.3.3 and 7.3.4). In Section 7.3.5, we further evaluate the fingerprint provision process, i.e., the performance of the immutable ledger. Finally, we conclude our evaluation in Section 7.4.

7.1. Implementation and Experimental Setup

We implemented Python-based prototypes for the collaborators, the information coordinator, and the access guards. For the immutable ledger, we utilize Quorum [101], an Ethereum [117] fork, which has been proven to support more than 2000 transactions per second [92]. For the evaluation, we relied on four Quorum nodes. In contrast to Ethereum, we use the Raft [118] consensus algorithm, which follows the proof-of-authority concept [119]. Since we can neither influence the performance of external storage providers nor cover all potential providers, we exclude external storage providers from our evaluation. We further implement our hybrid encryption scheme with the help of Charm [120] and the provided implementations of AES and the decentralized ABE scheme [99]. For the used digital signature derivation, i.e., for queries and fingerprints, we rely on the implementations provided by the eth-account Python library [121]. The information coordinator runs MongoDB [122], which serves as the database backend, while Apache 2.2 [123] handles all requests issued by the collaborators to forward them to our implementation via ModWSGI [124].

We conduct all measurements on a single server with two Intel XeonSilver 4116 CPUs, i.e., 12 cores and 24 threads each, as well as a total of 196 GB RAM. During our experiments, we run all entities and processes on the same machine. We repeat each measurement 20 times and report the standard deviation over these runs if not stated otherwise.

7.2. Supply Chain Model

To conduct a realistic evaluation, we rely on a real-world supply chain. The insights of the composition graph of the e.GO Life enable us to analyze PrivAccIChain's performance based on a realistic setting. We derive two evaluation models from the e.GO scenario that we originally introduced in Section 6. We refer to the first model as *base model* hereinafter. In this model, we represent each component of the e.GO Life as an individual product of the product flow

DAG, which corresponds to the creation of a *produce* record. Hereby, we deviate from the real structure and assume that each production step is executed by a unique company, such that we include ownership transfers represented by *trade* records between each production step. Therefore, our evaluation considers the worst case performance-wise.

As we visualize in Figure 7, the e.GO Life is assembled out of more than 90 pre-assembled components, such as the body, front doors, the rear axle, the battery, and the engine. Each of these components consists of multiple components themselves. In contrast to our previous evaluation [40], the resulting tree structure is broader and features a more irregular branching behavior. However, it comprises only 10 instead of 17 levels (cf. Figure 6). Due to the detailed insights into the e.GO Life's supply chain, we assess a maximum depth of 10 as more realistic. In particular, the products on this (final) level already correspond to basic components, such as screws. Thus, for our base model, we obtain a tree with 2222 nodes and 2221 edges as shown in Figure 6. The longest path in this tree, highlighted by orange nodes in the figure, corresponds to a decomposition of the aluminum spaceframe. While the inner nodes on levels three and four mainly correspond to pre-assembled body parts, levels nine and ten represent basic components, such as screws, nuts, or rivets – the so-called "c-parts". Hence, the presented scenario already covers fine-granular product decompositions and underlines the real-world relevance of our evaluation model.

On the levels three and four, leaves either represent basic components, such as screws or brackets, or they represent different pre-assembled components such as electronic control units (ECUs) produced by external suppliers. Thus, we derive a second, *extended model* from the e.GO Life's composition graph to also take the production steps of these pre-assembled parts into account. To this end, we append seven full ternary trees of depth three to each leaf on these two levels. This addition results in 280 added nodes for each original leaf on these levels, as we showcase using the example of the *Window Lifter Cover* in Figure 7. For illustration, we highlight multiple exemplary leaves that we extended with red circles in Figure 6. These tree structures approximate the pre-production steps for each component. Since the actual composition graph already consists of basic parts on these levels, our model represents an overapproximation. The resulting extended model consists of 133 822 nodes and 133 821 edges. Consequentially, this extended model, originating from a real-world composition graph, is comparable to our previous evaluation [40] in scale.

7.3. Performance Measurements

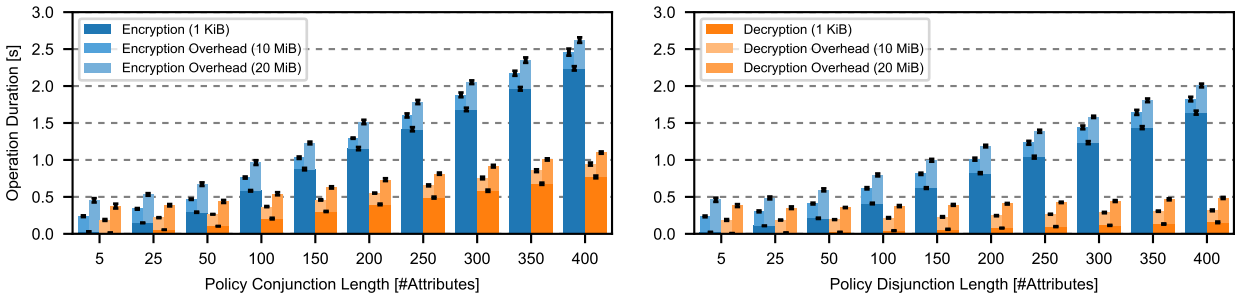
In the following, we present the results of our comprehensive performance evaluation that bases on a real-world, purchasable product, the e.GO Life. We evaluate different operations of our architecture separately to ascertain PrivAccIChain's performance regarding computational overhead introduced by the encryption scheme, durations of common operations such as information provision and retrieval under different configurations, and transaction throughput and overhead of the immutable ledger. As part of our evaluation, we utilize both supply chain models that we introduced in Section 7.2. In particular, we first analyze the ABE performance, which is part of our hybrid encryption scheme, in Section 7.3.1. Second, in Section 7.3.2, we look into the data record creation and update performance before focusing on the information retrieval in Section 7.3.3. Afterward, in Section 7.3.4, we investigate the tracking and tracing behavior of our architecture. Finally, we detail the performance of the immutable ledger in Section 7.3.5.

7.3.1. ABE Performance

The provision, updating, and retrieval of data records usually include encryption or decryption processes that enable our fine-granular access control and ensure data privacy (**G3**). Consequentially, the performance of these operations is influenced by the performance of our hybrid encryption scheme. To determine the extent of this influence, we provide a dedicated analysis of the performance of our utilized encryption scheme.

ABE Bootstrapping. To give an impression of the overhead for setting up a multi-authority ABE environment, we measured the required setup time. This step does not influence individual encryption and decryption operations. It consists of the generation of *global parameters*, the setup of key authorities, which corresponds to the generation of *attribute secret keys*, and the generation of *attribute private keys* that are issued to the collaborators and represent their decryption capabilities. For the generation of attribute private keys, a key authority, which corresponds to an access guard (cf. Section 5.2), needs the corresponding attribute secret key.

The one-time generation of the global parameters takes 5.17 ms $\pm$ 0.17 ms. Similarly, the derivation of an attribute secret key requires 5.97 ms $\pm$ 0.19 ms. Each access guard has to derive an individual secret key for each attribute managed by this access guard. Finally, generating an attribute private key takes 6.37 ms $\pm$ 0.25 ms per key. An attribute private key has to be generated every time an access guard issues the respective attribute to a unique collaborator. While



(a) Performance implications of conjuncted ABE attributes

(b) Performance implications of disjuncted ABE attributes

Figure 8: The used AES keys are embedded as an ABE ciphertext policy with a varying number of ABE attributes. While the number of ABE attributes linearly influences the cryptographic operations, the AES-based payload encryption times are independent of the underlying ABE policy. Overall, the decryption runtime outperforms the encryption.

the global parameter and attribute secret key generation times are neglectable due to their rare occurrence, issuing attribute private keys can easily be parallelized arbitrarily. Considering our decentralized access guard architecture, the burden for each access guard introduced by these non-repetitive operations is still limited, and therefore feasible.

Conjunctions and Disjunctions. To evaluate the performance of encryption and decryption operations with our hybrid encryption scheme, we consider the total number of access guards and attributes, the number of attributes utilized in an ABE policy, the payload size, and the policy composition, i.e., the influence of attribute conjunctions and disjunctions. Since neither the total number of access guards nor the total number of available attributes influence the performance of these operations due to the properties of the underlying ABE scheme [99], we do not have to adjust our measurements with respect to these factors. In Figure 8, we visualize how the remaining three factors (number of utilized attributes, payload size, and policy composition) influence the encryption and decryption performance. We cover policies with attribute conjunctions in Figure 8a and attribute disjunctions in Figure 8b, respectively. In particular, a policy conjunction of length n corresponds to a policy of the form $\bigwedge_{i=1}^{n} a_i$, where each a_i is a unique attribute. Similarly, a policy disjunction of length n equals $\bigvee_{i=1}^{n} a_i$.

Payload Sizes. For both policy compositions, we compare the times for encrypting and decrypting payloads (of different sizes 1 KiB, 10 MiB, and 20 MiB) with varying policy lengths. Encryption and decryption durations grow linearly (correlation coefficient ~0.9999) with increasing policy length. Furthermore, we observe that increased payload sizes lead to a linear overhead, whereby this overhead is independent of the utilized policy as it only affects the symmetric encryption of our hybrid scheme. Encrypting 20 MiB instead of 1 KiB leads to increased encryption and decryption durations by 318 ms ± 19 ms. The decryption process is generally about twice as fast as the corresponding encryption operation. Since the overhead for symmetric cryptography is identical for encryption and decryption, the shorter runtimes stem from faster ABE decryption.

Policy Length. Our measurements reveal the influence of the access policy design on the performance of our hybrid encryption scheme. While more complex policies allow for finer granularity regarding the access control, and further contribute to higher collusion resistance (cf. Section 5.4), larger policies have a significant impact on the observed performance. A policy consisting of 25 attribute conjunctions results in an encryption duration of 148 ms ± 4 ms for 1 KiB payload size, whereas increasing the number of utilized attributes to 400 results in an encryption duration of 2.23 s ± 29 ms. We anticipate that 50 attributes per policy are sufficient in most use cases. As we elaborate in Section 6, 10 production and assembly steps are sufficient to assemble a complex product (e.g., automobiles) from basic components (e.g., screws). In such cases, 50 available attributes suffice to express the access rights of all collaborators along the supply chain with decent levels of policy-level collusion resistance (cf. Section 5.4). Taking also into account that encryption and decryption are performed by the collaborators, i.e., the load is distributed to several entities, we assess the performance of our hybrid encryption system as reasonable and sufficient even for large-scaled supply chains.

Based on these insights on PrivAccIChain's encryption scheme's performance, we evaluate those operations that perform data record encryption and decryption in the following sections.

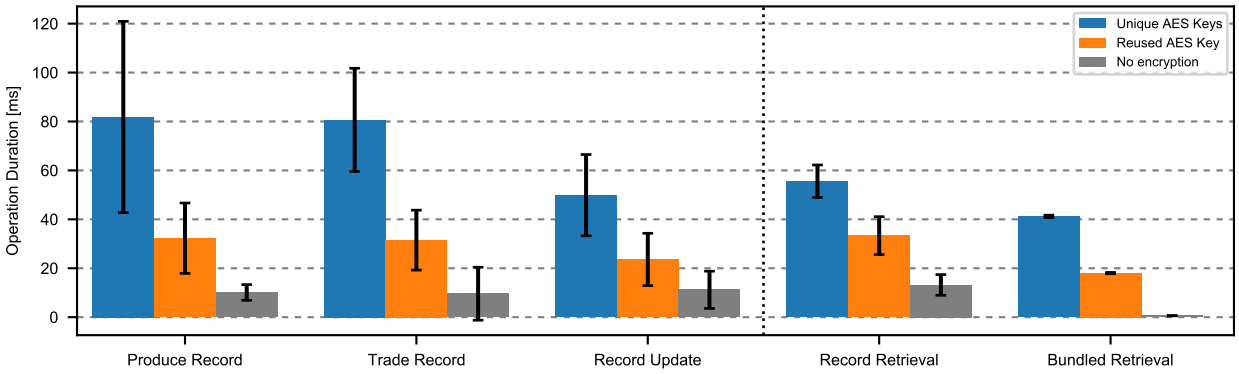


Figure 9: Operation Durations for Data Record Provision, Updates, and Retrieval for three encryption scenarios. For each operation, the cryptographic operations performed by the collaborators represent the dominating processing influence. A utilization of bundled record retrievals allows a tunable trade-off between retrieval latency and throughput.

7.3.2. Data Record Creation and Updates

Our next performance evaluation covers the creation and updates of data records. These operations represent the production processes as well as the corresponding ownership transfers. We evaluate both the durations for creating and updating individual data records as well as the resulting consequences with respect to our derived supply chain model (cf. Section 7.2). For each node in the respective DAG, we create a *produce* record with a payload size of ∼1 KiB, while we create a *trade* record for each edge. We explicitly apply the insertion of *trade* records between production steps conducted at e.GO, such that our evaluation represents a worst-case scenario regarding performance. Thus, we simulate extremely challenging future scenarios where all production steps are distributed across different companies.

Encryption Scenarios. Further, although we dedicatedly evaluated the performance of our encryption scheme, we evaluate its impact on the operational performance for data record creation and updates, as well as on the data record retrieval (cf. Section 7.3.3). To this end, we consider three different scenarios with respect to the application of encryption, (i) utilizing no encryption for neither the data record's payload nor for the tracking and tracing references (as a baseline), (ii) applying encryption for these three data record fields while reusing AES keys along the supply chain, which allows collaborators to cache the resulting ABE-encrypted keys and increases performance, and (iii) applying encryption with unique AES keys for each data record and even for different data record fields, i.e., we encrypt tracking and tracing references and the data record's payload with unique AES keys. With these encryption scenarios, we cover a baseline scenario representing the capabilities of the information coordinator, an optimized but realistic encryption scenario, and a worst-case scenario. We refer to these encryption scenarios during our evaluation of data record creation, updates, and retrieval, as well as for our evaluation of tracking and tracing. To increase the load on the information coordinator, we parallelize the creation and updating operations by using 30 processes. We now discuss our evaluation results regarding data record creation and updates, which we also visualize in Figure 9.

Record Creation. The creation of a single *produce* data record takes 81 ms ± 39 ms in the worst-case encryption scenario with unique AES keys over more than 400 000 samples. This creation time includes the encryption time of 70 ms ± 38 ms, i.e., encryption is the dominating influence here. Similarly, the creation of *trade* records takes 80 ms ± 21 ms. The higher deviation for *produce* records results from varying branching conditions inside the underlying tree, such that different data records require a different number of encryption iterations to cover the varying number of references to previous production steps. Reusing AES keys reduces the required time to 32 ms ± 14 ms for *produce* records and 31 ms ± 12 ms for *trade* records, i.e., we achieve a speedup of more than 50 % while still encrypting all data records. For all creation operations in these two encryption scenarios, the time the information coordinator takes to process the request is less than 5 ms on average, which underlines that the primary load is on the collaborators, indicating the absence of scalability issues (cf. **G5**). This statement is further backed by the record creation times for unencrypted records, where the creation of *produce* records only takes 10 ms ± 3 ms and 9 ms ± 10 ms for *trade* records.

Record Updates. Apart from the creation of data records, existing data records are updated during the production process to provide updated tracking references or additional product information. In our considered models, each product, represented by a *produce* record, is utilized by exactly one other product. The final product, i.e., the e.GO

Life, is the only exception here (cf. Figure 6). Consequentially, data record updates correspond to the addition of exactly one tracking reference per operation. Hereby, each *produce* and each *trade* record (except for the final *produce* record) is updated exactly once. A single record update takes 49 ms ± 16 ms in the worst-case encryption scenario over a sample size of more than 800 000. With reused AES keys, an update only takes 23 ms ± 10 ms, which is further reduced to 11 ms ± 7 ms when providing unencrypted reference updates. Handling data record updates takes the information coordinator slightly more time compared to the creation of data records, since updating data records additionally requires the information coordinator to load the requested data record, update the affected record fields, and to maintain a consistent version history.

Registering all production steps and ownership transfers, including the required data record updates, takes 43 s in our basic supply chain model (2222 production steps) and 31 min for our extended supply chain model (133 822 production steps) in the worst-case encryption scenario. A single record creation operation takes ≈ 7 ms on average for both supply chain models, whereby the operations for the basic model are slightly slower than for the extended model as different creation steps exhibit stronger dependencies on each other due to the lower number of total production steps. Compared to transaction validation durations—corresponding to record creation in our architecture—of 4 ms to 8 ms in ProductChain [38], we hence achieve similar performance as we include all encryption operations in our measurements. In a real deployment, these operations are never executed without delay, nor are they submitted by a single entity as each collaborator interacts with the information coordinator individually after processing (and shipping) the product. Hence, the achieved total insertion runtime is more than appropriate for both architectures. Next, we shift to the performance evaluation of data record retrieval.

7.3.3. Data Record Retrieval

Data records stored at the side of the information coordinator can be requested by authorized collaborators. Besides the retrieval of information for individual products and related production information, tracking and tracing require iterative record retrieval for a large number of data records. Before we cover these special applications in the next section, we first evaluate the performance of data record retrieval in general with respect to the introduced encryption scenarios. Since collaborators might need to request multiple data records independently of tracking and tracing, e.g., in case of larger deliveries, we support request bundling in PrivAccIChain. Instead of issuing a single query for each desired data record, collaborators can request multiple hundreds of data records with a single query. While this design increases the complexity for this query, it reduces the overhead resulting from request signatures and round trip times, such that we expect a significant performance benefit for the effective retrieval time per data record.

We show the evaluation results for both, single record queries and bundled queries with 100 requested records per query, in Figure 9. Requesting a single record takes 55 ms ± 6 ms with unique AES keys, 33 ms ± 7 ms with reused AES keys, and 13 ms ± 4 ms for unencrypted data records. The effective time per requested record with bundling is significantly lower. In particular, the effective times average at 41 ms ± 5 ms, 18 ms ± 0.28 ms, and respectively 0.59 ms±0.06 ms for the three different encryption scenarios. Especially for unencrypted data records, bundling results in a significant speedup and underlines the potential benefits of bundling for tracking and tracing. Since retrieval times do not exceed 60 ms, we generally assess these record retrieval times as performant and real-world applicable.

7.3.4. Tracking and Tracing

Based on our evaluation of retrieval of data records in general and the advantages of bundled queries, we now evaluate the data record retrieval with respect to tracking and tracing, i.e., multi-hop information retrieval, in the context of our e.GO supply chain scenario (cf. Section 2.3). In PrivAccIChain, we implement both tracking and tracing as iterative data record retrieval, where a collaborator requests those records that are referenced in the respective tracking or tracing fields. We provide the pseudo code for the tracing algorithm in Listing 1 in Appendix Appendix A. These iterative requests enable tracking and tracing without the necessity for the information coordinator to have access to the tracking and tracing reference fields itself, which contributes to achieving comprehensive data privacy (**G3**).

Tracking & Tracing Setting. For the models that we derived from the e.GO graph, tracing corresponds to a complete traversal of the product flow graph, beginning at the unique sink, i.e., the e.GO Life as the final product. In the base scenario, a full trace of the automobile reveals 2222 production steps with 2221 ownership transfers represented by *trade* records. For the extended model, a full trace consists of 133 822 *produce* record retrievals and 133 821 *trade* records, respectively. Tracking, on the other hand, corresponds to following a single path from an arbitrary node in the product flow graph towards the final product. For our evaluation, we choose a leaf at the maximum depth, i.e., on

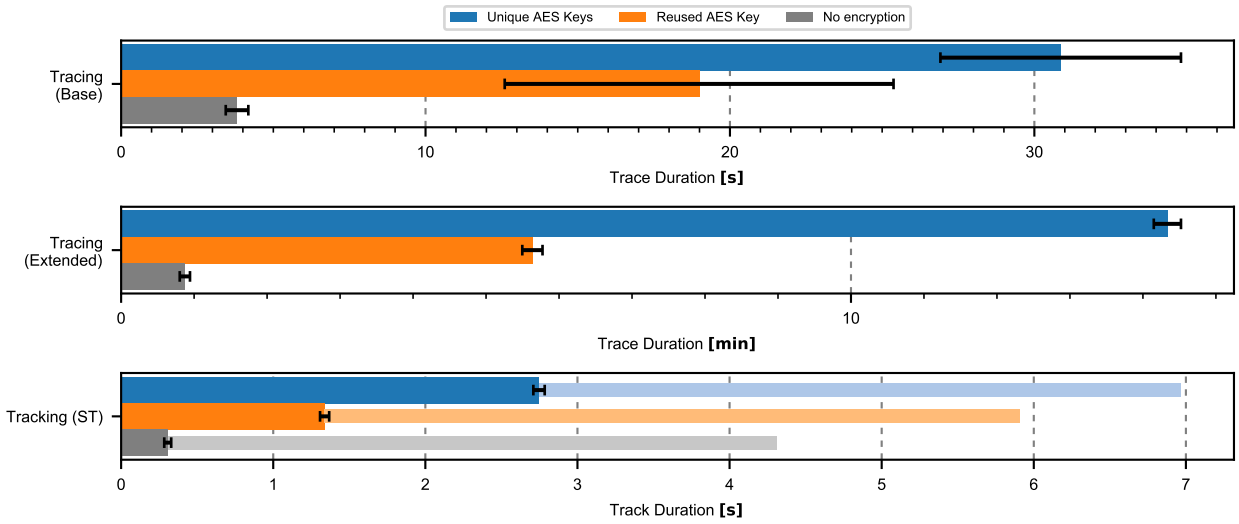


Figure 10: Durations for performing a full trace of the e.GO Life in the base model as well as in the extended model. Similar to data record provision and retrieval, tracking and tracing durations are dominated by decryption operation durations, which is underlined by the significant differences between unencrypted data records and those records that utilize unique encryption keys for each operation. As the longest path is identical for both models, the provided tracking duration applies to both models. Since tracking corresponds to following a single path in the present model, we state the duration when using a single-threaded (ST) process. The fainted bars indicate the overhead of multi-processing.

level 10, for tracking. These leaves correspond to basic components, such as screws. Thus, we evaluate the worst-case tracking scenario with respect to the supply chain models of the e.GO Life.

Tracing Performance. We parallelize the tracing procedure with 30 processes and allow bundles of size 500 for encrypted data records. For our baseline measurement, we use a single process with 10 threads and request bundling of up to 2000 data records for unencrypted data records. While using multiple processes increases the computation capabilities for decrypting the data records, the resulting synchronization overhead leads to a performance loss with unencrypted data records that require significantly fewer computation capabilities at the collaborators.

For both models, we visualize the tracing evaluation results in Figure 10. Please note that we utilize different scales due to the significant difference in the number of individual production steps. Performing a full trace of the final product, i.e., the e.GO Life, in the basic supply chain model takes 30.87 s ± 3.95 s for uniquely encrypted data records, 18.99 s ± 6.39 s for data records encrypted with reused AES keys, and 3.81 s ± 0.37 s for unencrypted data records over 20 runs for each encryption scenario. For the extended model with 133 822 *produce* records, a full trace requires 14:20 min ± 11 s with unique AES keys, 5:38 min ± 8 s for reused AES keys, and 0:52 min ± 4 s for unencrypted tracing information. Taking the effective time for unencrypted data records of 0.59 ms with a bundle size of 100 as a baseline (cf. Section 7.3.3), 267 643 data records can theoretically be retrieved in 157 s utilizing a single thread. The execution of our iterative, collaborator-driven tracing implementation takes up to 20 iterations to reveal the longest path with all corresponding *produce* and *trade* records. Here, the utilizations of 10 threads results in a tracing time of 52 s and fulfills our expectations as it underlines the computational capabilities of the information coordinator. During all tracing processes, neither the information coordinator's database nor the webserver were operated near their maximum load, i.e., these components do not constitute performance bottlenecks.

We consider the necessity for a full trace of a complete product flow graph a rare event, e.g., in case of a car accident, a place crash, or food poisoning. Hence, the tracing durations remain reasonable even in the worst-case encryption scenario with unique encryption keys for each data record. Since the retrieval of all data records is supported in a minute or less, increasing the computational capacities at the collaborator's side suffices to reduce the tracing durations, since most of the tracing time is spent for decrypting the retrieved data records. Parallelizing the tracing process increases the performance even further as different subtrees can be traversed independently, allowing for distributed and horizontally scalable tracing implementations.

Tracking Performance. Besides tracing, we identify tracking as a second essential application (cf. Section 2.3).

Table 2: Comparison of fingerprint provision performance for dedicated transactions per fingerprint and bundling of 200 fingerprints per transaction. We provide the transaction sizes, effective sizes per fingerprint, preparation times per transaction and fingerprints as well as the transaction and fingerprint throughput for both, dedicated transactions and fingerprint bundles. While using fingerprint bundles reduces the transaction throughput and leads to higher preparation duration, the resulting fingerprint throughput is increased compared to dedicated transactions per fingerprint.

Transaction Scheme	TX size	TX size/FP	Prep./TX	Prep./FP	TX/s	FP/s
Dedicated TX	265 B	265 B	5.50 ms	5.50 ms	~741	~741
Fingerprint Bundle	32 429 B	~126 B	230 ms	1.15 ms	~6.45	~1289

In the present models, the longest path consists of 10 *produce* and 9 *trade* records for both, the basic and the extended supply chain model. As for the tracing times, we present the tracking duration for all three encryption scenarios in Figure 10. Since parallelization is not beneficial when following a single path, we provide the tracking performance of a single-threaded process instead. To highlight the necessity for careful decisions regarding process and thread numbers, we additionally provide the overhead resulting from parallelizing tracking with 30 synchronized processes. For all encryption scenarios, the information coordinator load is insignificant. In the worst-case encryption scenario, tracking takes 2.75 s $\pm$ 0.03 s, while reusing AES keys reduces the tracking time to 1.33 s $\pm$ 0.03 s. For unencrypted tracking information, the complete tracking takes 0.31 s $\pm$ 0.02 s.

Since every record query only requests a single record, we cannot achieve any benefits from bundling for tracking. The average multi-processing overhead is higher than 4 s for all encryption scenarios. While tracking in all scenarios is supported in a reasonable time, the performance is limited by the collaborator-driven tracking implementation that does not require the information coordinator to have access to tracking and tracing references. Although we assess tracking and tracing references as sensitive data, information coordinator-driven tracking and tracing can improve the performance significantly and can be considered a reasonable alternative for use cases with established trust. Similar to the protocol in ProductChain [38], the information coordinator could resolve and follow tracking and tracing references, which mitigates the necessity for multiple iterations. The resulting privacy implications are, however, to consider.

Performance Comparison. Despite the significant design differences between related approaches, we can assess the overall performance of our architecture as comparable to ambitious approaches such as ProductChain [38]. Tracing or tracking in ProductChain takes 37 ms [38] for 35 transactions (which corresponds to 35 data records). Opposed to our architecture, data records are stored on the immutable ledger and contain unencrypted tracing and tracking information, i.e., tracing and tracking do not require iterative queries by collaborators. Tracking or tracing following a single path consisting of 35 data records without applied encryption takes 0.52 s in our architecture. Since the retrieval of 100 data records takes 59 ms with bundled requests, we achieve comparable performance for identical supply chain structures, despite the differences in the underlying architectures. Due to the increased performance for record retrieval when using bundled requests, we are confident that information coordinator-driven tracing and tracking, i.e., when iterations are performed by the information coordinator instead of the clients, can achieve even lower operation durations. In such a scenario, the performance is limited by the query performance of the information coordinator's database. An extensive evaluation of MongoDB's performance in the context of supply chain traceability [125] suggests that the information coordinator would need less than 50 ms for iteratively providing 35 data records.

Altogether, we achieve sufficient performance for rarely required full traces as well as for the tracking of individual products regardless of the evaluated model. Since both operations mainly put load on the retrieving collaborator, the imposed computational requirements mitigate the risk for excessive information revelation by collaborators. Increasing the decryption capabilities of collaborators allows them to achieve a significant speedup, which we underline by providing our (unencrypted) baseline measurements that outline potentially achievable performance. Similarly, our architecture provides sufficiently short processing for revealing single paths or carefully chosen parts of the supply chain, such that both applications are feasible in real-world supply chains.

7.3.5. Immutable Ledger

During data record creation and updates, collaborators and the information coordinator agree on deterministic fingerprints for the immutable ledger (cf. Section 5.6). Since the provision of fingerprints is no time-critical operation, the information coordinator creates the respective transactions separately from the immediately processed data record creation and update operations. Hence, we independently evaluate the fingerprint creation and its storage requirements.

Fingerprint Bundling. To store the fingerprint on the immutable ledger, the information coordinator issues a

signed transaction containing the data record's ID and the fingerprint along with the request type, the collaborator address, the record type and the data record's version as additional information. The combined size of these parameters adds up to 124 B, whereby the total size of the corresponding transaction totals at 265 B. Since the transaction overhead does not grow linearly with payload size, bundling multiple fingerprints into a single transaction reduces the overhead per fingerprint. A single transaction bundling 200 fingerprints and the corresponding metadata has a size of 32 429 B, which corresponds to an effective size of ∼162 B per fingerprint compared to 265 B, i.e., bundling achieves a significant reduction of transaction size overhead. Due to transaction size limitations, we cannot bundle an arbitrary number of fingerprints into a single transaction. Bundling fingerprints results in larger transactions and a (theoretical) higher delay for fingerprint provision, since the information coordinator needs to aggravate the respective number of fingerprints. Therefore, these implications must be kept in mind when enabling a bundling mechanism.

We summarize the resulting transaction sizes as well as the duration for transaction preparation and the effective transaction throughput in Table 2. When bundling fingerprints, we only write ∼6.45 transactions per second (TX/s) to the ledger. In contrast, we achieve ∼741 TX/s for dedicated per-fingerprint transactions. However, by this measure we increase the payload throughput from ∼741 fingerprints per second (FP/s) to ∼1289 FP/s. Although transaction throughput and, consequentially, fingerprint throughput can be further increased by utilizing more verifying nodes [126], we identify the immutable ledger as the bottleneck of our architecture. Hence, we propose three additional, immutable ledger-specific optimizations to improve our architecture's scalability (**G5**).

Transaction Optimization. To simplify the fingerprint verification process and to provide a convenient interface even for manual inspection, we designed a smart contract [127] handling the fingerprints to include redundant information, e.g., the collaborator's address, the request type as well as the data record's version. By removing the redundant information, we can achieve a smaller fingerprint transaction payload while still maintaining all desired accountability (**G1**) and verifiability (**G2**) features. This change would allow for more fingerprints per bundle, resulting in a higher fingerprint throughput at the cost of increased verification complexity.

Meta Fingerprints. Similarly, the introduction of meta fingerprints can improve the scalability at the cost of increased verification complexity. Instead of storing each record's fingerprint on the ledger, the information coordinator calculates a meta fingerprint over multiple data record fingerprints. To achieve a storage reduction of 99 % on the immutable ledger, the information coordinator can combine 100 28 B fingerprints into a single 28 B meta fingerprint. However, the verification process requires the collaborators to retrieve all combined fingerprints, resulting in increased complexity. Advanced concepts, such as Merkle trees [128], can mitigate the resulting fingerprint verification overhead. For example, instead of chaining n fingerprints for verification, only $\log_2 n$ specific hashes are needed.

Alternative Ledgers and Sharding. Finally, choosing a different blockchain technology can increase the ledger's performance in general. While Quorum provides a promising performance with additional features, such as smart contracts and private transactions, other scenarios can contain different requirements regarding the ledger's minimum set of properties. Sharding [129] promises to increase performance by splitting the ledger into several local chains that apply independent transaction verification [130]. For instance, RapidChain [130] claims a potential throughput of 7380 TX/s. For large and distributed supply chains, utilizing several completely detached ledgers with a deterministic selection mechanism (to identify the responsible ledger for a specific data record) can increase performance as well. The utilization of the data record ID as the single unique identifiers eases such a deterministic selection and allows for horizontal scalability of the immutable ledger.

While the current transaction rates could be considered a bottleneck of our implementation [131], PrivAccIChain's design itself is oblivious of the concrete underlying blockchain. Hence, improving the fingerprint throughput for supply chains with high demands is a simple and minor adaption. Three protocol-specific modifications can further improve the capabilities of the immutable ledger significantly at only moderate costs (e.g., increased verification overhead).

7.4. Evaluation Discussion

In our evaluation, we considered the real-world supply chain of the e.GO Life, an electric urban city car. Besides the base model that we derived from the e.GO composition graph in Figure 6, we extended our evaluation to verify PrivAccIChain's scalability by covering more than 130 000 production steps and ownership transfers and including pre-production steps for more complex parts such as electronic control units (cf. Section 7.2). For both models, we further considered three different encryption scenarios (cf. Section 7.3.2) to analyze the impact of our hybrid encryption scheme and the resulting trade-off between performance, privacy preservation, and transparency. Despite significantly

different impacts of the three encryption scenarios, i.e., privacy preservation levels, on data record provision, update, and retrieval operation performance (cf. Figure 9), we assess the performance for these operations as appropriate, as all operations take less than 100 ms on average. Here, the overhead for most cryptographic operations is measured at the collaborator, i.e., the system scales well with an increasing number of operations and collaborators. Since our architecture allows for different encryption schemes that entail different levels of privacy preservation (**G3**), we achieve a tunable trade-off between privacy and performance for the possibility of automated and accountable multi-hop information provision and retrieval. Nevertheless, as all operations are in the magnitude of a few milliseconds, our architecture is well-suited for supply chains where production and transportation processes often take hours, days, or even weeks in contrast to our evaluation scenario where production and transportation are considered as instantaneous.

Further, we ascertain that the underlying supply chain structure, i.e., the branching behavior and depth of the DAG, does not have a major impact on the performance for tracing. Compared to our previous evaluation [40] that considered longer paths, yet sparser branching behavior, we observe linear scaling regarding the tracing durations. Hence, we conclude that for supply chains with n nodes, which exceeds the number of utilized processes p by several magnitudes (i.e., $n \gg p$), only n has a significant impact on the tracing duration. Finally, although we identified the immutable ledger as PrivAccIChain's bottleneck regarding its performance and scalability, suitable techniques, such as meta fingerprints, sharding, and the utilization of multiple independent ledgers, or a different underlying blockchain suffice for appropriate scalability even in large-scale supply chains (cf. Section 7.3.5).

Overall, we conclude that PrivAccIChain provides satisfying performance for real-world supply chains. Our results are in line with other current, promising supply chain approaches [38] (cf. Section 7.3.4). With the presented flexibility of PrivAccIChain, i.e., its customizability regarding accountability, verifiability, and privacy at the cost of computational complexity and storage requirements, we extend existing work and provide a dynamic, yet powerful architecture. Next, we elaborate on PrivAccIChain's security to also underline its well-considered security properties.

8. Security Discussion

Besides performance, the security of PrivAccIChain is of eminent relevance as the security features are essentially influencing the architecture's ability to reliably protect business secrets and sensitive information. In Section 8.1, we first introduce PrivAccIChain's utilized building block, before we discuss twelve attack vectors that represent a potential threat in Section 8.2. Finally, in Section 8.3, we present a summary of all discussed attack vectors.

8.1. Security of Utilized Building Blocks

PrivAccIChain utilizes several cryptographic and technological concepts such as attribute-based encryption, symmetric encryption, and elliptic curve cryptography (ECC) [132] for its design. These building blocks are combined with blockchain technology that provides a consensus between distrusting parties and a tamperproof log of recorded information. We rely on the individual security of these concepts and technologies as well as their respective implementations, namely AES [133], the utilized CP-ABE scheme [99], Ethereum [117] and Quorum [101], as well as all utilized libraries, e.g., the Charm cryptography framework [120]. Additionally, we require that the underlying software and hardware components work according to their specification and do not introduce unnecessary attack surfaces. Finally, we consider the effects of external factors, such as long-term power outages, as out of scope and assume that PrivAccIChain is operational. Therefore, we focus on design-specific attack vectors in the following.

8.2. Attack Vector Analysis

With the security of our utilized building blocks in mind, we now discuss potential attack vectors against PrivAccIChain. Hereby, we comparably analyze the likelihood of the respective attack as well as the severity in case of a successful attack. For our assessment, we partially rely on findings from related work regarding individual attack vectors to derive their qualitative rating of severity and likelihood with respect to the design and properties of PrivAccIChain. In total, we present twelve attack vectors and order them according to a decreasing severity.

Entity Collusion. **Entities:** Information Coordinator, Access Guards **Severity:** ● **Likelihood:** ◑

Entity collusion refers to attempts of multiple entities within our architecture to gain access to information in violation of established access policies. Since information is stored in encrypted data records at the side of the information coordinator, the information coordinator has to be involved in the collusion to provide the encrypted data. Alternatively,

a collaborator can distribute the data record after requesting it from the information coordinator. However, in this case, the collaborator could also decrypt the record itself. The decryption of the data record requires access to the symmetric key, which can be derived with the help of attribute private keys. This situation results in two potential scenarios.

First, collaborators are in possession of attribute private keys issued by one or multiple access guards. If one collaborator legitimately has all attribute private keys that are required for decryption, this collaborator has access to the data record following the access policy, such that no illegitimate access occurs. Due to the collusion resistance of the underlying ABE scheme, joining attribute private keys of different collaborators will not lead to increased decryption capabilities, such that illegitimate access is prevented [99].

Second, those access guards that control the issuance of attribute private keys of those attributes that are required for decryption can collude to obtain decryption capabilities for the given data record. In contrast to a collusion of multiple collaborators, this behavior cannot be prevented by our architecture. However, due to the distribution of control over attributes by introducing multiple access guards, a tunable collusion resistance can be achieved by increasing the number of access guards that have to collude. The risk of entity collusion can be mitigated by appropriately designing access control policies (cf. Section 5.4).

Intentional Data Distribution. **Entities:** Information Coordinator, Collaborators **Severity:** ● **Likelihood:** ○

Besides the collusion of multiple entities, data records might get into the possession of unauthorized parties following an intentional distribution. If encrypted data records are distributed, no information is leaked as the payload is protected through our hybrid encryption scheme. Additionally, a collaborator with legitimate payload access can also reveal the decrypted information. Consequentially, our architecture cannot prevent the distribution of decrypted information after the respective information has left the system. However, access logs at the information coordinator can help to identify the responsible collaborator.

Data Loss, Manipulation, and Deletion. **Entities:** Information Coordinator **Severity:** ● **Likelihood:** ○

With the information coordinator as a conceptually centralized entity, the collaborators trust this entity to store data records persistently. Thus, the existence and correctness of these data records depend on the information coordinator's correct behavior. Hence, unintentional data loss as well as intentional manipulation or deletion of data records represent attack vectors against our architecture. The risk of data loss can be mitigated by storing multiple replicas of the data records at different locations as well as by performing regular backups. Using multiple information coordinator instances, run by different operators, each maintaining one or multiple replicas of each data record, significantly reduces the risk of data loss. Similarly, intentional manipulations and deletion of data records are detectable by collaborators due to the fingerprints stored on the immutable ledger. While a malicious information coordinator could also delete and manipulate existing backups and replicas, utilizing multiple information coordinator instances represents an appropriate countermeasure against these attack vectors in general and can further improve the scalability.

Majoritarian Illegitimate Behavior. **Entities:** Inf. Coordinator, Collaborators, Access Guards, Ledger Nodes **Severity:** ● **Likelihood:** ○

Our design relies on the assumption that most entities behave according to their specification and do not collude to manipulate, delete, or distribute data. Therefore, majoritarian illegitimate behavior can impact PrivAccIChain as well as the security and privacy mechanisms in place.

By distributing the control over attribute key issuance to multiple access guards, encrypting data records, storing proofs of existence on the immutable ledger, and allowing multiple instances of the information coordinator, our architecture provides several countermeasures against entity misbehavior and collusion. The level of resilience against illegitimate behavior is tunable at the cost of increased complexity and decreased performance. Nevertheless, our architecture is vulnerable to a scenario where all access guards, for instance, collude with the information coordinator, i.e., where a majority of central entities colludes or misbehaves. In such a case, neither data privacy nor its existence can be guaranteed by our architecture. While this attack represents a significant threat, we rate the likelihood as very low and counterintuitive, because collaborators will not rely on the respective instance of PrivAccIChain anymore.

Information Fraud. **Entities:** Collaborators **Severity:** ◕ **Likelihood:** ◕

As soon as information has entered the system, it is protected against manipulation and deletion with the help of decentralization and fingerprints. However, our architecture cannot guarantee the correctness and validity of provided information due to payload encryption and the flexible data format. Thus, collaborators can conduct information fraud attacks by submitting syntactical correct data records without valuable or semantically correct information. This attack explicitly covers both manipulated information and random data. Since data records are usually encrypted by the providing collaborator, verification of the viability of information is only possible after retrieval and decryption by another collaborator. Our architecture comprises the detached judge for on-demand conflict resolution. For example, informa-

tion fraud can be punished with financial penalties. However, further mechanisms, such as an integrated reputation system (cf. TrustChain [59]) or the utilization of tamperproof sensor data [105], can provide additional resilience.

Request Pattern Analyses. **Entities:** Information Coordinator, Collaborators **Severity:** ◑ **Likelihood:** ◔

Besides attacks that try to manipulate or delete data records and get illegitimate access to a data record's information, gaining knowledge on supply chain relationships and interactions is possible by analyzing the interactions between collaborators and the information coordinator. Although we encrypt the communication payload, metadata, such as the origin and the destination of a request as well as request patterns (e.g., request frequencies and volume), can be observed without granted access. By design, the information coordinator has access to these types of information. Access to data records of a collaborator provided by another collaborator indicates a direct or indirect business relationship, for instance. Further, the frequency of data record provision might indicate production volumes, problems, or utilization.

As potential countermeasures, collaborators can use multiple accounts, i.e., they appear as different collaborators to the information coordinator at different network locations. Alternatively, active deception by providing obfuscated data records, i.e., data records without relation to products and production processes as well as access to these data records for randomized collaborators represents an appropriate countermeasure. Although we cannot prevent knowledge gain by analyzing request patterns, our architecture offers possibilities to impede these analyses if needed or desired.

Key Leakage. **Entities:** Collaborators, Access Guards **Severity:** ◑ **Likelihood:** ○

Our system achieves accountability and verifiability as well as data privacy with the help of cryptographic building blocks (i.e., AES and ABE) that require several cryptographic keys. In particular, PrivAccIChain requires (i) attribute private keys (ii) attribute secret keys, and (iii) authentication private keys to be kept secret by the respective entities. Leakage of these keys to external or internal parties can compromise the security and data privacy, such that key leakage, including targeted attacks to steal one or multiple keys, represents another credible attack vector.

Attribute private keys are issued to collaborators by one or multiple access guards. They are required to decrypt the symmetric key used for payload encryption (cf. Section 5.3). If an attacker gets access to one or multiple attribute private keys, he gains the respective decryption capabilities. Due to the collusion resistance of the underlying ABE scheme, joining keys from multiple collaborators does not lead to increased decryption capabilities [99]. Since the information coordinator implements an additional layer of access control, the respective attacker usually has no access to encrypted data records, i.e., an additional entity must be involved for information revelation. Here, the detection of leaked attribute private keys is important, such that the affected attributes can be replaced.

Similarly, the leakage of attribute secret keys, i.e., keys held by access guards and required to issue new attribute private keys to collaborators, represents only a minor threat to data privacy. If an attacker gets access to one or multiple attribute secret keys of a single access guard, this attacker can issue the associated attribute private keys for respective decryption capabilities. As for the attribute private key leakage, the additional level of access control implemented by the information coordinator prevents access to encrypted data records such that the security impact is marginal. Furthermore, an appropriate policy design (cf. Section 5.4), which enforces that attributes issued by different access guards are needed for decryption, prevents any information revelation even if data records are leaked. As soon as the leakage of attribute secret keys has been detected, the affected attributes need to be replaced to maintain security.

Finally, all entities authenticate themselves to others with the help of digital signatures that require a corresponding public and private key pair. This key is used for authentication at the information coordinator, during the fingerprint deviation and signing process as well as for signing transactions on the immutable ledger. In case such a private key is compromised, an attacker can use the key to act on behalf of the collaborator that usually possesses the private key, i.e., he can provide and request data records on behalf of this collaborator. However, possession of the authentication private key does not lead to increased decryption capabilities because the appropriate ABE attributes are missing, i.e., the attacker cannot decrypt any retrieved data records. With the help of request origin determination or request logs, authentication leakages can be detected and the associated access privileges can be revoked.

Altogether, key leakage represents a significant attack vector with only limited severity due to our multi-layered and decentralized access control scheme. The implementation of advanced mechanisms, such as time-interval attributes [134], can further reduce the implications of key leakage.

Immutable Ledger Manipulation. **Entities:** Ledger Nodes **Severity:** ◑ **Likelihood:** ○

In PrivAccIChain, we provide a proof of existence as well as a proof of originality for each data record with the help of an immutable ledger that stores fingerprints of these data records. To provide accountability, a current and consistent state of the ledger is required. Our implementation utilizes Quorum with the Raft [118] consensus protocol, which essentially requires more than 50 % of all nodes to be active and to confirm a transaction. Consequentially,

Table 3: Summary of potential attack vectors against PrivAccIChain. For each attack, we state the estimated likelihood and severity in case of a successful attack as well as all implemented and optional (highlighted in italics) countermeasures. We rate likelihood and severity as low ○, medium-low ◔, medium ◑, medium-high ◕, or high ●.

Attack	Severity	Likelihood	Countermeasures
Entity Collusion	●	◑	Multiple Information Coordinator Instances, Multiple Access Guards, *External Supervision*
Intentional Data Distribution	●	○	Request Logging, *Financial Penalties*
Data Loss, Manipulation, and Deletion	●	○	Data Backups, Multiple Information Coordinator Instances, Fingerprints, Proof of Existence
Majoritarian Illegitimate Behavior	●	○	Multiple Access Guards, Multiple Information Coordinator Instances
Information Fraud	◕	◕	Record Fingerprints, Financial Penalties, *Reputation System* [59, 139], *IoT Sensor Data* [105]
Request Pattern Analyses	◑	◔	Multiple Accounts per Participant, *Active Obfuscation*
Key Leakage	◑	○	Encryption combined with Access Control, Time-Interval Attributes [134]
Immutable Ledger Manipulation	◑	○	Careful Node Operator Selection
Identity Spoofing	◔	○	Authentication via Signatures
DoS & DDoS Attacks	◔	○	*Preemptive and Reactive DoS Avoidance* [136, 137]
Collaborator Discrimination	○	◑	ABE-based Access Policies
Data Leakages	○	◑	Data Record Payload Encryption

manipulation of the immutable ledger requires an attacker to take control of a majority of ledger nodes, i.e., to perform a 51 %-attack [38]. The success of such an attack is unlikely given an appropriate number of ledger nodes and a careful selection of operators. Even if such an attack succeeds, it is detectable and potential manipulations can be reverted.

Identity Spoofing. **Entities:** All Entities **Severity:** ◔ **Likelihood:** ○

To obtain access to data records, an attacker can attempt to spoof the identity of another entity in our architecture. We utilize digital signatures to authenticate each request between collaborators and the information coordinator, such that a successful identity spoofing attack requires the attacker to possess the private keys of its spoofed identity. An aspect that we covered before (cf. key leakage). Additionally, PrivAccIChain prevents replay attacks by including timestamps and random nonces in each request. Finally, brute force attacks on the 256 bit private key [117] are still computationally infeasible [135]. Besides, the security of the utilized key material can be configured as needed.

DoS & DDoS Attacks. **Entities:** All Entities **Severity:** ◔ **Likelihood:** ○

Besides specialized attacks against our architecture, denial of service (DoS) and Distributed DoS (DDoS) attacks target systems with the objective to hinder users from utilizing the attacked infrastructure in its intended way. Both, internal and external parties can perform these types of attacks. Since internal entities are well-known, respective DoS attacks are identifiable. Thus, we can generalize internal attacks to the harder detectable and preventable external attacks, where appropriate countermeasures are presented in past [136, 137] and current research [138]. Resource accounting as well as pattern and anomaly detection represent preventive and reactive methods. Due to our distributed architecture and the recording of usually time-uncritical information, DoS attacks only pose a minor threat to our system.

Collaborator Discrimination. **Entities:** Collaborators **Severity:** ○ **Likelihood:** ◑

Since data records can be stored for a longer period of time before all collaborators with authorized access request these records, selective discrimination of individual collaborators during key distribution is possible depending on the utilized distribution mechanism. With a classical public-private-key encryption scheme, the symmetric key is encrypted individually for each recipient. Thus, the attacker can encrypt an invalid key for a specific collaborator such that this collaborator will not be able to decrypt the data record. Such an attack might not be uncovered for a long period due to a potentially severely delayed data record retrieval. ABE-based key distribution prohibits such attacks by design, since the data providing collaborator encrypts the symmetric key only once, such that all recipients retrieve the same symmetric key. Consequentially, either all authorized collaborators can decrypt the data record or none. Furthermore, the distribution of an invalid key is detected as soon as a single collaborator tries to decrypt the data record, such that the provision of invalid decryption material can be detected more quickly.

Data Leakages. **Entities:** Information Coordinator, Collaborators **Severity:** ○ **Likelihood:** ◑

Similar to the intentional data distribution, data records might be revealed by an unintentional distribution or as a result of an attack against the information coordinator. However, although we assess the likelihood of external attacks higher than for intentional data distribution, external attacks usually do not reveal the required decryption keys. Hence, attackers gain no access to valuable information since the payload is still encrypted and protected by ABE policies.

8.3. Conclusion of our Security Discussion

We identified twelve potential attack vectors against PrivAccIChain and analyzed their severity as well as their respective likelihoods. Furthermore, we discussed several implemented as well as optional countermeasures. In Table 3, we summarize all attack vectors and respective countermeasures.

Overall, we categorize information fraud as the most likely attack against PrivAccIChain, since collaborators might expect competitive advantages by providing invalid, made-up, or tampered information. The potential severity of information fraud is well-known [45], which underlines our assessment of information fraud being a critical attack vector. Although information fraud might be detectable upon manual or even automated inspection, advanced features such as the inclusion of trusted sensors [105] or a reliable reputation system [59, 139] paired with legal penalties for misbehaving collaborators are required to further mitigate the risk of information fraud. Due to the tunable resilience against entity collusion of our architecture by utilizing multiple access guards, we assess the remaining attack vectors as unlikely and rather uncritical. Thus, we conclude that PrivAccIChain provides appropriate data protection capabilities as well as a tunable trade-off between transparency and data privacy along supply chains.

9. Conclusion and Future Work

We proposed PrivAccIChain as a design that enables privacy-preserving and accountable multi-hop transparency even in dynamic supply chains. In Section 9.1, we first summarize the findings of our literature review before concluding the presentation of our proposed architecture PrivAccIChain. Finally, we discuss future work in Section 9.2.

9.1. Conclusion

Even though a multi-hop collaborations between business partners along a supply chain positively affects business performance, profits, and customer satisfaction [7, 11], their implementation requires advanced solutions to mitigate negative impacts. For instance, emerging paradigms such as the Internet of Production [13] steer toward short-term business relationships without prior trust, which must be accounted for. Recent approaches that address associated challenges follow different paradigms. While active mediation or decentralized business process monitoring [91] enable mutual agreement on multi-hop production processes, other approaches [19, 23, 29, 38, 62] propose product modeling to enable multi-hop product and production transparency. However, as highlighted in our extensive literature review (cf. Section 3), these approaches either address specific supply chains, e.g., food supply chains [23, 38, 75], require trust between businesses [62], or do not provide appropriate data privacy or needed scalability [21, 29].

Despite the valuable contributions of these approaches, we identified a lack of approaches combining all desired properties (cf. Section 2.4). Thus, we presented PrivAccIChain as a novel approach that achieves a tunable trade-off between data transparency and privacy and provides accountability and verifiability by distributing control over critical system components to multiple entities. PrivAccIChain provides reasonable performance that suffices for real-world supply chains as our comprehensive performance evaluation of PrivAccIChain's functionalities demonstrates. Since we base our evaluation on insights into the realistic supply chain at e.GO Mobile AG and its e.GO Life automobile, our evaluation surpasses evaluations of other approaches not only regarding scale, but also regarding its realism and real-world impact. We complemented our evaluation with a detailed analysis of potential attack vectors as well as a discussion of appropriate mitigations and deduce that PrivAccIChain provides adequate and use case-adaptable security standards.

However, we want to emphasize that aspects regarding information accountability beyond our architecture's boundaries require further research, which we elaborate on in the subsequent discussion of future work.

9.2. Future Work

Generally, we can categorize future work into three groups. First, a fine-tuning of certain design aspects can help to support use case-specific needs in PrivAccIChain. For example, attribute commercialization promises to develop new business models as information stored as part of PrivAccIChain can be offered for sale by the participating collaborators. Similarly, a time-based ABE encryption scheme [134] allows PrivAccIChain to also operate in scenarios with challenging data privacy needs where transparency might be rated as less important. On a related note, various mechanisms can extend the functionality of PrivAccIChain. Reputation systems, which have been previously proposed

for the use in blockchain-backed supply chains [59], can support dynamic supply chain environments as companies can source additional information about yet unknown potential business partners and suppliers.

Second, supporting users with a selection of setup options for PrivAccIChain can improve its readiness for real-world deployments. For example, building on top of our security discussion (cf. Section 8), PrivAccIChain would benefit from clearly defined concepts that propose how to configure all relevant security parameters during operation. Here, especially a rate-limiting system for interacting with the information coordinator comes to mind. This feature could potentially be further linked to the previously discussed reputation system. Furthermore, the ABE policy design could be formalized for different scenarios to allow for a simple setup of PrivAccIChain across diverse supply chains in the future. As we stated before (cf. Section 5.4), designing policies is a complex, yet crucial aspect of our architecture.

Third, further research is required at the boundaries of our architecture, i.e., several orthogonal research questions remain. A major issue is the trustworthy and reliable sensing of information [3, 105]. PrivAccIChain can only provide accountability and verifiability after the data entered the system. Hence, ensuring data correctness is an important research direction for utilizing our architecture to the best extent. For example, both a clear linking between physical and digital objects [23, 140, 141] as well as a trustworthy recording [105] is required. Even though related work [31] frequently identifies these aspects as open challenges, no completely satisfying and universally suitable approaches are available. To get a broader overview of open adoption challenges, Gonczol et al. [31] compiled an extensive list that they extracted from related work and industry.

Acknowledgments

This work is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC-2023 Internet of Production – 390621612. The authors are further grateful for the fruitful discussions on supply chains with Philipp Niemietz from the Laboratory for Machine Tools and Production Engineering (WZL) at RWTH Aachen University. We would also like to thank the anonymous reviewers and the associate editor for their valuable feedback and comments.

References

[1] D. M. Lambert, M. C. Cooper, Issues in Supply Chain Management, Industrial Marketing Management 29 (1) (2000) 65–83. doi:10.1016/S0019-8501(99)00113-3.

[2] M. Christopher, Logistics & Supply Chain Management, 5th Edition, FT Press, 2016.

[3] K. Wüst, A. Gervais, Do you need a Blockchain?, in: Proceedings of the 2018 Crypto Valley Conference on Blockchain Technology (CVCBT '18), IEEE, 2018, pp. 45–54. doi:10.1109/CVCBT.2018.00011.

[4] D. Berdik, S. Otoum, N. Schmidt, D. Porter, Y. Jararweh, A Survey on Blockchain for Information Systems Management and Security, Information Processing & Management 58 (1). doi:10.1016/j.ipm.2020.102397.

[5] J. Pennekamp, E. Buchholz, Y. Lockner, M. Dahlmanns, T. Xi, M. Fey, C. Brecher, C. Hopmann, K. Wehrle, Privacy-Preserving Production Process Parameter Exchange, in: Proceedings of the 36th Annual Computer Security Applications Conference (ACSAC '20), ACM, 2020, pp. 510–525. doi:10.1145/3427228.3427248.

[6] J. Pennekamp, M. Henze, S. Schmidt, P. Niemietz, M. Fey, D. Trauth, T. Bergs, C. Brecher, K. Wehrle, Dataflow Challenges in an *Internet* of Production: A Security & Privacy Perspective, in: Proceedings of the ACM Workshop on Cyber-Physical Systems Security & Privacy (CPS-SPC '19), ACM, 2019, pp. 27–38. doi:10.1145/3338499.3357357.

[7] T. M. Simatupang, R. Sridharan, The Collaborative Supply Chain, The International Journal of Logistics Management 13 (1) (2002) 15–30. doi:10.1108/09574090210806333.

[8] T. Moyaux, B. Chaib-draa, S. D'Amours, Information Sharing as a Coordination Mechanism for Reducing the Bullwhip Effect in a Supply Chain, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 37 (3) (2007) 396–409. doi:10.1109/TSMCC.2006.887014.

[9] V. Dedeoglu, A. Dorri, R. Jurdak, R. A. Michelin, R. C. Lunardi, S. S. Kanhere, A. F. Zorzo, A Journey in Applying Blockchain for Cyberphysical Systems, in: Proceedings of the 2020 International Conference on COMmunication Systems & NETworkS (COMSNETS '20), IEEE, 2020, pp. 383–390. doi:10.1109/COMSNETS48256.2020.9027487.

[10] M. L. Fisher, A. Raman, A. S. McClelland, Rocket Science Retailing Is Almost Here–Are You Ready?, Harvard Business Review 78 (4) (2000) 115–123.

[11] D. M. Lambert, M. A. Emmelhainz, J. T. Gardner, Building successful logistics partnerships, Journal of Business Logistics 20 (1) (1999) 165–181.

[12] R.-D. Leon, R. Rodríguez-Rodríguez, P. Gómez-Gasquet, J. Mula, Business process improvement and the knowledge flows that cross a private online social network: An insurance supply chain case, Information Processing & Management 57 (4). doi:10.1016/j.ipm.2020.102237.

[13] 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, K. Wehrle, Towards an Infrastructure Enabling the Internet of Production, in: Proceedings of the 2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS '19), IEEE, 2019, pp. 31–37. doi:10.1109/ICPHYS.2019.8780276.

[14] J. Pennekamp, M. Dahlmanns, L. Gleim, S. Decker, K. Wehrle, Security Considerations for Collaborations in an Industrial IoT-based Lab of Labs, in: Proceedings of the 3rd IEEE Global Conference on Internet of Things (GCIoT '19), IEEE, 2019. doi:10.1109/GCIoT47977.2019.9058413.

[15] L. Gleim, J. Pennekamp, M. Liebenberg, M. Buchsbaum, P. Niemietz, S. Knape, A. Epple, S. Storms, D. Trauth, T. Bergs, C. Brecher, S. Decker, G. Lakemeyer, K. Wehrle, FactDAG: Formalizing Data Interoperability in an Internet of Production, IEEE Internet of Things Journal 7 (4) (2020) 3243–3253. doi:10.1109/JIOT.2020.2966402.

[16] M. Henze, The Quest for Secure and Privacy-preserving Cloud-based Industrial Cooperation, in: Proceedings of the 2020 IEEE Conference on Communications and Network Security (CNS '20), IEEE, 2020, proceedings of the 6th International Workshop on Security and Privacy in the Cloud (SPC '20). doi:10.1109/CNS48642.2020.9162199.

[17] Everledger Limited, Do You Know Your Diamond?, https://diamonds.everledger.io/ (2019 (accessed December 27, 2019)).

[18] IBM, IBM Announces Major Blockchain Collaboration with Dole, Driscoll's, Golden State Foods, Kroger, McCormick and Company, McLane Company, Nestlé, Tyson Foods, Unilever and Walmart to Address Food Safety Worldwide, https://www-03.ibm.com/press/us/en/pressrelease/53013.wss (2017 (accessed May 1, 2020)).

[19] S. A. Abeyratne, R. Monfared, Blockchain ready manufacturing supply chain using distributed ledger, International Journal of Research in Engineering and Technology 5 (9). doi:10.15623/ijret.2016.0509001.

[20] S. Wang, D. Li, Y. Zhang, J. Chen, Smart Contract-Based Product Traceability System in the Supply Chain Scenario, IEEE Access 7 (2019) 115122–115133. doi:10.1109/ACCESS.2019.2935873.

[21] H. M. Kim, M. Laskowski, Toward an ontology-driven blockchain design for supply-chain provenance, Intelligent Systems in Accounting, Finance and Management 25 (1) (2018) 18–27. doi:10.1002/isaf.1424.

[22] K. Toyoda, P. T. Mathiopoulos, I. Sasase, T. Ohtsuki, A Novel Blockchain-Based Product Ownership Management System (POMS) for Anti-Counterfeits in the Post Supply Chain, IEEE Access 5 (2017) 17465–17477. doi:10.1109/ACCESS.2017.2720760.

[23] F. Tian, An agri-food supply chain traceability system for China based on RFID & blockchain technology, in: Proceedings of the 2016 13th International Conference on Service Systems and Service Management (ICSSSM '16), IEEE, 2016. doi:10.1109/ICSSSM.2016.7538424.

[24] M. Pincheira Caro, M. S. Ali, M. Vecchio, R. Giaffreda, Blockchain-based traceability in Agri-Food supply chain management: A practical implementation, in: Proceedings of the 2018 IoT Vertical and Topical Summit on Agriculture-Tuscany (IoT Tuscany '18), IEEE, 2018. doi:10.1109/IOT-TUSCANY.2018.8373021.

[25] R. Nayak, A. Singh, R. Padhye, L. Wang, RFID in textile and clothing manufacturing: technology and challenges, Fashion and Textiles 2 (1) (2015) 1–16. doi:10.1186/s40691-015-0034-9.

[26] H. Wu, Z. Li, B. King, Z. Ben Miled, J. Wassick, J. Tazelaar, A Distributed Ledger for Supply Chain Physical Distribution Visibility, Information 8 (4). doi:10.3390/info8040137.

[27] FarmaTrust, Saving Lives — Building Trust, White paper, FarmaTrust (2018).

[28] P. Helo, Y. Hao, Blockchains in operations and supply chains: A model and reference implementation, Computers & Industrial Engineering 136 (2019) 242–251. doi:10.1016/j.cie.2019.07.023.

[29] M. Westerkamp, F. Victor, A. Küpper, Blockchain-Based Supply Chain Traceability: Token Recipes Model Manufacturing Processes, in: Proceedings of the 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), IEEE, 2018, pp. 1595–1602, proceedings of the 2018 IEEE International Conference on Blockchain (Blockchain '18). doi:10.1109/Cybermatics_2018.2018.00267.

[30] S. Figorilli, F. Antonucci, C. Costa, F. Pallottino, L. Raso, M. Castiglione, E. Pinci, D. Del Vecchio, G. Colle, A. R. Proto, et al., A Blockchain Implementation Prototype for the Electronic Open Source Traceability of Wood along the Whole Supply Chain, Sensors 18 (9). doi:10.3390/s18093133.

[31] P. Gonczol, P. Katsikouli, L. Herskind, N. Dragoni, Blockchain Implementations and Use Cases for Supply Chains-A Survey, IEEE Access 8 (2020) 11856–11871. doi:10.1109/ACCESS.2020.2964880.

[32] V. Bryan, Aerospace suppliers look to blockchain for parts tracking, https://www.reuters.com/article/us-aerospace-blockchain/aerospace-suppliers-look-to-blockchain-for-parts-tracking-idUSKBN1I32AW (2018 (accessed May 1, 2020)).

[33] M. J. Casey, P. Wong, Global Supply Chains Are About to Get Better, Thanks to Blockchain, Harvard Business Review (2017) 2–6.

[34] N. Hackius, M. Petersen, Blockchain in Logistics and Supply Chain: Trick or Treat?, in: Proceedings of the Hamburg International Conference of Logistics (HICL '17), Vol. 23, epubli, 2017, pp. 3–18. doi:10.15480/882.1444.

[35] T. Hardin, D. Kotz, Amanuensis: Information provenance for health-data systems, Information Processing & Management 58 (2). doi:10.1016/j.ipm.2020.102460.

[36] B. Putz, M. Dietz, P. Empl, G. Pernul, EtherTwin: Blockchain-based Secure Digital Twin Information Management, Information Processing & Management 58 (1). doi:10.1016/j.ipm.2020.102425.

[37] C. Esposito, M. Ficco, B. B. Gupta, Blockchain-based authentication and authorization for smart city applications, Information Processing & Management 58 (2). doi:10.1016/j.ipm.2020.102468.

[38] S. Malik, S. S. Kanhere, R. Jurdak, ProductChain: Scalable Blockchain Framework to Support Provenance in Supply Chains, in: Proceedings of the 2018 IEEE 17th International Symposium on Network Computing and Applications (NCA '18), IEEE, 2018. doi:10.1109/NCA.2018.8548322.

[39] Q. Chen, G. Srivastava, R. M. Parizi, M. Aloqaily, I. Al Ridhawi, An incentive-aware blockchain-based solution for internet of fake media things, Information Processing & Management 57 (6). doi:10.1016/j.ipm.2020.102370.

[40] J. Pennekamp, L. Bader, R. Matzutt, P. Niemietz, D. Trauth, M. Henze, T. Bergs, K. Wehrle, Private Multi-Hop Accountability for Supply Chains, in: Proceedings of the 2020 IEEE International Conference on Communications Workshops (ICC Workshops '20), IEEE, 2020, proceedings of the 1st Workshop on Blockchain for IoT and Cyber-Physical Systems (BIoTCPS '20). doi:10.1109/ICCWorkshops49005.2020.9145100.
[41] H. Håkansson, International Marketing and Purchasing of Industrial Goods: An Interaction Approach, Wiley, 1982.
[42] H. Zhang, Vertical information exchange in a supply chain with duopoly retailers, Production and Operations Management 11 (4) (2002) 531–546. doi:10.1111/j.1937-5956.2002.tb00476.x.
[43] B. B. Flynn, B. Huo, X. Zhao, The impact of supply chain integration on performance: A contingency and configuration approach, Journal of Operations Management 28 (1) (2010) 58–71. doi:10.1016/j.jom.2009.06.001.
[44] V. R. Aprilio, B. Bergmans, Implementation of Blockchain for Increasing Traceability at VehGro Supply Chain, Diponegoro Journal of Accounting 9 (4).
[45] The Aircraft Accident Investigation Board / Norway, Report on the Convair 340/580 LN-PAA aircraft accident North of Hirtshals, Denmark on September 8, 1989 (1993).
[46] I. Omar, M. Debe, R. Jayaraman, K. Salah, M. Omar, J. Arshad, Blockchain-based Supply Chain Traceability for COVID-19 PPE, TechRxiv (2020).
[47] J. Pennekamp, P. Sapel, I. B. Fink, S. Wagner, S. Reuter, C. Hopmann, K. Wehrle, M. Henze, Revisiting the Privacy Needs of Real-World Applicable Company Benchmarking, in: Proceedings of the 8th Workshop on Encrypted Computing & Applied Homomorphic Cryptography (WAHC '20), HomomorphicEncryption.org, 2020, pp. 31–44. doi:10.25835/0072999.
[48] Y. Tribis, A. El Bouchti, H. Bouayad, Supply Chain Management based on Blockchain: A Systematic Mapping Study, MATEC Web of Conferences 200, proceedings of the International Workshop on Transportation and Supply Chain Engineering (IWTSCE '18). doi:10.1051/matecconf/201820000020.
[49] Y. Wang, M. Singgih, J. Wang, M. Rit, Making sense of blockchain technology: How will it transform supply chains?, International Journal of Production Economics 211 (2019) 221–236. doi:10.1016/j.ijpe.2019.02.002.
[50] Y. Wang, J. H. Han, P. Beynon-Davies, Understanding blockchain technology for future supply chains: a systematic literature review and research agenda, Supply Chain Management 24 (1) (2019) 62–84. doi:10.1108/SCM-03-2018-0148.
[51] K. S. Hald, A. Kinra, How the blockchain enables and constrains supply chain performance, International Journal of Physical Distribution & Logistics Management 49 (4) (2019) 376–397. doi:10.1108/IJPDLM-02-2019-0063.
[52] M. Pournader, Y. Shi, S. Seuring, S. C. L. Koh, Blockchain applications in supply chains, transport and logistics: a systematic review of the literature, International Journal of Production Research 58 (7) (2019) 2063–2081. doi:10.1080/00207543.2019.1650976.
[53] P. Scully, M. Höbig, Exploring the impact of blockchain on digitized Supply Chain flows: A literature review, in: Proceedings of the 2019 6th International Conference on Software Defined Systems (SDS '19), IEEE, 2019, pp. 278–283. doi:10.1109/SDS.2019.8768573.
[54] P. Katsikouli, A. S. Wilde, N. Dragoni, H. Høgh-Jensen, On the benefits and challenges of blockchains for managing food supply chains, Journal of the Science of Food and Agriculture.
[55] S. Jabbar, H. Lloyd, M. Hammoudeh, B. Adebisi, U. Raza, Blockchain-enabled supply chain: analysis, challenges, and future directions, Multimedia Systemsdoi:10.1007/s00530-020-00687-0.
[56] J. Al-Jaroodi, N. Mohamed, Blockchain in Industries: A Survey, IEEE Access 7 (2019) 36500–36515. doi:10.1109/ACCESS.2019.2903554.
[57] J. Wang, P. Wu, X. Wang, W. Shou, The outlook of blockchain technology for construction engineering management, Frontiers Of Engineering Management 4 (1) (2017) 67–75. doi:10.15302/J-FEM-2017006.
[58] T. K. Mackey, G. Nayyar, A review of existing and emerging digital technologies to combat the global trade in fake medicines, Expert Opinion on Drug Safety 16 (5) (2017) 587–602. doi:10.1080/14740338.2017.1313227.
[59] S. Malik, V. Dedeoglu, S. S. Kanhere, R. Jurdak, TrustChain: Trust Management in Blockchain and IoT supported Supply Chains, in: Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain '19), IEEE, 2019, pp. 184–193. doi:10.1109/Blockchain.2019.00032.
[60] D. G. Glover, J. Hermans, Improving the Traceability of the Clinical Trial Supply Chain, Applied Clinical Trials 26 (12) (2017) 36–38.
[61] H.-C. Pfohl, M. Gomm, Supply chain finance: optimizing financial flows in supply chains, Logistics Research 1 (3-4) (2009) 149–161. doi:10.1007/s12159-009-0020-y.
[62] S. Appelhanz, V.-S. Osburg, W. Toporowski, M. Schumann, Traceability system for capturing, processing and providing consumer-relevant information about wood products: system solution and its economic feasibility, Journal of Cleaner Production 110 (2016) 132–148. doi:10.1016/j.jclepro.2015.02.034.
[63] Project Provenance Ltd., Provenance, https://www.provenance.org/technology (2013 (accessed January 26, 2020)).
[64] T. Kelepouris, K. Pramatari, G. Doukidis, RFID-enabled traceability in the food supply chain, Industrial Management & Data Systems 107 (2) (2007) 183–200. doi:10.1108/02635570710723804.
[65] Z. Pang, J. Chen, Z. Zhang, Q. Chen, L. Zheng, Global fresh food tracking service enabled by wide area wireless sensor network, in: Proceedings of the 2010 IEEE Sensors Applications Symposium (SAS '10), IEEE, 2010, pp. 6–9. doi:10.1109/SAS.2010.5439425.
[66] F. Tian, A supply chain traceability system for food safety based on HACCP, blockchain & Internet of things, in: Proceedings of the 2017 14th International Conference on Service Systems and Service Management (ICSSSM '17), IEEE, 2017. doi:10.1109/ICSSSM.2017.7996119.
[67] S. S. M. Bahrudin, M. I. Illyas, M. I. Desa, Tracking and tracing technology for halal product integrity over the supply chain, in: Proceedings of the 2011 International Conference on Electrical Engineering and Informatics (ICEEI '11), IEEE, 2011. doi:10.1109/ICEEI.2011.6021678.
[68] S. K. Palaniswamy, S. A. A. Balamurugan, A. Kumar, Implementation of a web-based real-time temperature monitoring of shellfish caches using wireless sensor networks, in: Proceedings of the 2008 International Conference on Computing, Communication and Networking (ICCCN '08), IEEE, 2008, pp. 1–6. doi:10.1109/ICCCNET.2008.4787691.
[69] P. M. Goodrum, M. A. McLaren, A. Durfee, The application of active radio frequency identification technology for tool tracking on con-

struction job sites, Automation in Construction 15 (3) (2006) 292–302. doi:10.1016/j.autcon.2005.06.004.
[70] L. Chen, L. Chen, S. Jordan, Y.-K. Liu, D. Moody, R. Peralta, R. Perlner, D. Smith-Tone, Report on Post-Quantum Cryptography, NIST IR 8105 (2016). doi:10.6028/NIST.IR.8105.
[71] Z. Gao, L. Xu, L. Chen, X. Zhao, Y. Lu, W. Shi, CoC: A Unified Distributed Ledger Based Supply Chain Management System, Journal of Computer Science and Technology 33 (2) (2018) 237–248. doi:10.1007/s11390-018-1816-5.
[72] Peer Ledger Inc., Peer Ledger, https://peerledger.com/ (2016 (accessed May 1, 2020)).
[73] Ambrosus Team, Ambrosus, White paper, Ambrosus Technologies GmbH (2018).
[74] N. Maouriyan, A. G. A. Krishna, AQUACHAIN - Water Supply-Chain management using Distributed Ledger Technology, in: Proceedings of the 2019 3rd International Conference on Computing and Communications Technologies (ICCCT '19), IEEE, 2019, pp. 204–207. doi:10.1109/ICCCT2.2019.8824945.
[75] Y. Fernando, M. R. Darun, A. Z. Abideen, D. N. Ibrahim, M. Tieman, F. Mohamad, Adoption of Blockchain Technology to Improve Integrity of Halal Supply Chain Management, IGI Global, 2020, pp. 2488–2496. doi:10.4018/978-1-7998-3473-1.ch172.
[76] K. Gao, Y. Liu, H. Xu, T. Han, Hyper-FTT: A Food Supply-Chain Trading and Traceability System Based on Hyperledger Fabric, in: Proceedings of the 2nd International Conference on Blockchain and Trustworthy Systems (BlockSys '19), Vol. 1156, Springer, 2019, pp. 648–661. doi:10.1007/978-981-15-2777-7_53.
[77] A. Tayal, A. Solanki, R. Kondal, A. Nayyar, S. Tanwar, N. Kumar, Blockchain-based efficient communication for food supply chain industry: Transparency and traceability analysis for sustainable business, International Journal of Communication Systemsdoi:10.1002/dac.4696.
[78] Z. Shahbazi, Y.-C. Byun, A Procedure for Tracing Supply Chains for Perishable Food Based on Blockchain, Machine Learning and Fuzzy Logic, Electronics 10 (1) (2021) 41. doi:10.3390/electronics10010041.
[79] MediLedger, MediLedger, https://www.mediledger.com/ (2017 (accessed May 1, 2020)).
[80] T. Bocek, B. B. Rodrigues, T. Strasser, B. Stiller, Blockchains everywhere - a use-case of blockchains in the pharma supply-chain, in: Proceedings of the 2017 15th IFIP/IEEE Symposium on Integrated Network and Service Management (IM '17), IEEE, 2017, pp. 772–777. doi:10.23919/INM.2017.7987376.
[81] PharmaTrace, PharmaTrace, https://www.pharmatrace.io/ (2017 (accessed May 1, 2020)).
[82] Guardtime, Efficient Supply-Chain Management, https://guardtime.com/health/efficient-supply-chain-management (2019 (accessed May 1, 2020)).
[83] C.-L. Chen, Y.-Y. Deng, C.-T. Li, S. Zhu, Y.-J. Chiu, P.-Z. Chen, An IoT-Based Traceable Drug Anti-Counterfeiting Management System, IEEE Access 8 (2020) 224532–224548. doi:10.1109/ACCESS.2020.3036832.
[84] S. Tahir, J. A. Hussien, A Traceable and Reliable Electronic Supply Chain System Based on Blockchain Technology, UHD Journal of Science and Technology 4 (2) (2020) 132–140. doi:10.21928/uhdjst.v4n2y2020.pp132-140.
[85] OpenPort, OpenPort, https://openport.com/ (2015 (accessed May 1, 2020)).
[86] M. Lammi, Project SmartLog: blockchain in logistics, https://smartlog.kinno.fi/articles/project-smartlog-blockchain-logistics (2018 (accessed May 1, 2020)).
[87] Shipchain Inc., The ShipChain Ecosystem, https://docs.shipchain.io/docs/intro.html (2019 (accessed May 1, 2020)).
[88] X. Li, F. Lv, F. Xiang, Z. Sun, Z. Sun, Research on Key Technologies of Logistics Information Traceability Model Based on Consortium Chain, IEEE Access 8 (2020) 69754–69762. doi:10.1109/ACCESS.2020.2986220.
[89] B. Rakic, T. Levak, Z. Drev, S. Savic, A. Veljkovic, First Purpose-Built Protocol for Supply Chains Based Blockchain, White paper, OriginTrail (2017).
[90] ISO/IEC JTC 1/SC 27, ISO/IEC 27001:2013, Standard (2013).
[91] I. Weber, X. Xu, R. Riveret, G. Governatori, A. Ponomarev, J. Mendling, Untrusted Business Process Monitoring and Execution Using Blockchain, in: Proceedings of the 14th International Conference on Business Process Management (BPM '16), Vol. 9850, Springer, 2016, pp. 329–347. doi:10.1007/978-3-319-45348-4_19.
[92] A. Baliga, I. Subhod, P. Kamat, S. Chatterjee, Performance Evaluation of the Quorum Blockchain Platform, arXiv:1809.03421 (2018).
[93] K. Gao, Y. Liu, H. Xu, T. Han, Design and implementation of food supply chain traceability system based on Hyperledger Fabric, International Journal of Computational Science and Engineering 23 (2) (2020) 185–193. doi:10.1504/IJCSE.2020.110547.
[94] P. Altmann, A. G. Abbasi, O. Schelén, K. Andersson, M. Alizadeh, Creating a Traceable Product Story in Manufacturing Supply Chains Using IPFS, in: Proceedings of the 2020 IEEE 19th International Symposium on Network Computing and Applications (NCA '20), IEEE, 2020. doi:10.1109/NCA51143.2020.9306719.
[95] C. Oham, R. A. Michelin, R. Jurdak, S. S. Kanhere, S. Jha, B-FERL: Blockchain based framework for securing smart vehicles, Information Processing & Management 58 (1). doi:10.1016/j.ipm.2020.102426.
[96] A. Sahai, B. Waters, Fuzzy Identity-Based Encryption, in: Proceedings of the 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT '05), Vol. 3494, Springer, 2005, pp. 457–473. doi:10.1007/11426639_27.
[97] V. Goyal, O. Pandey, A. Sahai, B. Waters, Attribute-based encryption for fine-grained access control of encrypted data, in: Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS '06), ACM, 2006, pp. 89–98. doi:10.1145/1180405.1180418.
[98] B. Waters, Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization, in: Proceedings of the 14th International Conference on Practice and Theory in Public Key Cryptography (PKC '11), Vol. 6571, Springer, 2011, pp. 53–70. doi:10.1007/978-3-642-19379-8_4.
[99] A. Lewko, B. Waters, Decentralizing Attribute-Based Encryption, in: Proceedings of the 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT '11), Vol. 6632, Springer, 2011, pp. 568–588. doi:10.1007/978-3-642-20465-4_31.
[100] H. Baniata, A. Anaqreh, A. Kertesz, PF-BTS: A Privacy-Aware Fog-enhanced Blockchain-assisted task scheduling, Information Processing & Management 58 (1). doi:10.1016/j.ipm.2020.102393.
[101] JP Morgan, Quorum, https://www.goquorum.com/ (2016).
[102] L. Bader, J. C. Bürger, R. Matzutt, K. Wehrle, Smart Contract-Based Car Insurance Policies, in: Proceedings of the 2018 IEEE Globecom Workshops (GC Wkshps '18), IEEE, 2018. doi:10.1109/GLOCOMW.2018.8644136.

[103] E. Wagner, A. Völker, F. Fuhrmann, R. Matzutt, K. Wehrle, Dispute Resolution for Smart Contract-based Two Party Protocols, in: Proceedings of the 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC '19), IEEE, 2019, pp. 422–430. doi:10.1109/BLOC.2019.8751312.
[104] J. Bethencourt, A. Sahai, B. Waters, Ciphertext-Policy Attribute-Based Encryption, in: Proceedings of the 2007 IEEE Symposium on Security and Privacy (SP '07), IEEE, 2007, pp. 321–334. doi:10.1109/SP.2007.11.
[105] J. Pennekamp, F. Alder, R. Matzutt, J. T. Mühlberg, F. Piessens, K. Wehrle, Secure End-to-End Sensing in Supply Chains, in: Proceedings of the 2020 IEEE Conference on Communications and Network Security (CNS '20), IEEE, 2020, proceedings of the 5th International Workshop on Cyber-Physical Systems Security (CPS-Sec '20). doi:10.1109/CNS48642.2020.9162337.
[106] J. Li, J. Wu, G. Jiang, T. Srikanthan, Blockchain-based public auditing for big data in cloud storage, Information Processing & Management 57 (6). doi:10.1016/j.ipm.2020.102382.
[107] Q. Zhao, S. Chen, Z. Liu, T. Baker, Y. Zhang, Blockchain-based privacy-preserving remote data integrity checking scheme for IoT information systems, Information Processing & Management 57 (6). doi:10.1016/j.ipm.2020.102355.
[108] J. Sydow, E. Schüßler, G. Müller-Seitz, Managing Inter-Organizational Relations: Debates and Cases, Macmillan International Higher Education, 2015.
[109] M. Kowalski, Management von Open-Innovation-Netzwerken, Springer, 2018. doi:10.1007/978-3-658-20907-0.
[110] T. Sturgeon, R. Florida, Globalization and Jobs in the Automotive Industry, MIT IPC Globalization Working Paper 01-003 (2000).
[111] J. Pennekamp, R. Matzutt, S. S. Kanhere, J. Hiller, K. Wehrle, The Road to Accountable and Dependable Manufacturing, under Review.
[112] A. Giampieri, J. Ling-Chin, Z. Ma, A. Smallbone, A. P. Roskilly, A review of the current automotive manufacturing practice from an energy perspective, Applied Energy 261. doi:10.1016/j.apenergy.2019.114074.
[113] T. Gartzen, F. Brambring, F. Basse, Target-oriented Prototyping in Highly Iterative Product Development, Procedia CIRP 51 (1) (2016) 19–23. doi:10.1016/j.procir.2016.05.095.
[114] M. Rossini, F. Ciarapica, D. Matt, P. R. Spena, A preliminary study on the changes in the Italian automotive supply chain for the introduction of electric vehicles, Journal of Industrial Engineering and Management 9 (2) (2016) 450–486. doi:10.3926/jiem.1504.
[115] T. J. Sturgeon, J. Van Biesebroeck, Global value chains in the automotive industry: an enhanced role for developing countries, International Journal of Technological Learning, Innovation and Development 4 (1–3) (2011) 181–205. doi:10.1504/IJTLID.2011.041904.
[116] R. C. Lamming, N. D. Caldwell, D. A. Harrison, W. Phillips, Transparency in Supply Relationships: Concept and Practice, Journal of Supply Chain Management 37 (3) (2001) 4–10. doi:10.1111/j.1745-493X.2001.tb00107.x.
[117] G. Wood, Ethereum: A Secure Decentralised Generalised Transaction Ledger, Yellow ppaer, Ethereum (2014).
[118] JP Morgan, Raft-based consensus for Ethereum/Quorum, https://docs.goquorum.com/en/latest/Consensus/raft/ (2019 (accessed January 15, 2020)).
[119] S. De Angelis, L. Aniello, F. Lombardi, A. Margheri, V. Sassone, PBFT vs Proof-of-Authority: Applying the CAP Theorem to Permissioned Blockchain, in: Proceedings of the 2nd Italian Conference on Cyber Security (ITASEC '18), Vol. 2058, CEUR Workshop Proceedings, 2017.
[120] J. A. Akinyele, C. Garman, I. Miers, M. W. Pagano, M. Rushanan, M. Green, A. D. Rubin, Charm: a framework for rapidly prototyping cryptosystems, Journal of Cryptographic Engineering 3 (2) (2013) 111–128. doi:10.1007/s13389-013-0057-3.
[121] Ethereum – GitHub, eth-account, https://github.com/ethereum/eth-account/ (2018).
[122] MongoDB Inc., MongoDB, https://www.mongodb.com (2009).
[123] The Apache Software Foundation, Apache HTTP Server Version 2.4 Documentation, https://httpd.apache.org/docs/2.4/en/ (2020).
[124] G. Dumpleton, Apache/mod_wsgi, https://github.com/GrahamDumpleton/mod_wsgi (2007).
[125] Y.-S. Kang, I.-H. Park, S. Youm, Performance Prediction of a MongoDB-Based Traceability System in Smart Factory Supply Chains, Sensors 16 (12). doi:10.3390/s16122126.
[126] M. Schäffer, M. di Angelo, G. Salzer, Performance and Scalability of Private Ethereum Blockchains, in: Proceedings of the International Conference on Business Process Management: Blockchain and Central and Eastern Europe Forum (BPM: Blockchain and CEE Forum '19), Vol. 361, Springer, 2019, pp. 103–118. doi:10.1007/978-3-030-30429-4_8.
[127] T. Hu, X. Liu, T. Chen, X. Zhang, X. Huang, W. Niu, J. Lu, K. Zhou, Y. Liu, Transaction-based classification and detection approach for Ethereum smart contract, Information Processing & Management 58 (2). doi:10.1016/j.ipm.2020.102462.
[128] R. C. Merkle, A Digital Signature Based on a Conventional Encryption Function, in: Proceedings of the 7th Conference on the Theory and Applications of Cryptographic Techniques (CRYPTO '87), Vol. 293, Springer, 1987, pp. 369–378. doi:10.1007/3-540-48184-2_32.
[129] J. C. Corbett, J. Dean, M. Epstein, A. Fikes, C. Frost, J. J. Furman, S. Ghemawat, A. Gubarev, C. Heiser, P. Hochschild, W. Hsieh, S. Kanthak, E. Kogan, H. Li, A. Lloyd, S. Melnik, D. Mwaura, D. Nagle, S. Quinlan, R. Rao, L. Rolig, Y. Saito, M. Szymaniak, C. Taylor, R. Wang, D. Woodford, Spanner: Google's Globally-Distributed Database, in: Proceedings of the 10th USENIX Conference on Operating Systems Design and Implementation (OSDI '12), USENIX Association, 2012, pp. 251–260.
[130] M. Zamani, M. Movahedi, M. Raykova, RapidChain: Scaling Blockchain via Full Sharding, in: Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (CCS '18), ACM, 2018, pp. 931–948. doi:10.1145/3243734.3243853.
[131] X. Xu, G. Sun, L. Luo, H. Cao, H. Yu, A. V. Vasilakos, Latency performance modeling and analysis for hyperledger fabric blockchain network, Information Processing & Management 58 (1). doi:10.1016/j.ipm.2020.102436.
[132] D. Hankerson, A. Menezes, S. Vanstone, Guide to Elliptic Curve Cryptography, Springer, 2011. doi:10.1007/b97644.
[133] M. J. Dworkin, E. B. Barker, J. R. Nechvatal, J. Foti, L. E. Bassham, E. Roback, J. F. Dray Jr., Advanced Encryption Standard (AES), NIST FIPS 197 (2001). doi:10.6028/NIST.FIPS.197.
[134] J. K. Liu, T. H. Yuen, P. Zhang, K. Liang, Time-Based Direct Revocable Ciphertext-Policy Attribute-Based Encryption with Short Revocation List, in: Proceedings of the 16th International Conference on Applied Cryptography and Network Security (ACNS '18), Vol. 10892, Springer, 2018, pp. 516–534. doi:10.1007/978-3-319-93387-0_27.
[135] S. Burnett, S. Paine, The RSA Security's Official Guide to Cryptography, McGraw-Hill, 2001.
[136] P. Ferguson, D. Senie, Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing, IETF RFC 2827 (2000).

[137] J. Mirkovic, P. Reiher, A taxonomy of DDoS attack and DDoS defense mechanisms, ACM SIGCOMM Computer Communication Review 34 (2) (2004) 39–53. doi:10.1145/997150.997156.
[138] J. Rangasamy, L. Kuppusamy, G. Krishnan, Velmurugan, Evaluation of puzzle-enabled proxy-assisted denial-of-service protection for web services, International Journal of Information and Computer Security 9 (1–2) (2017) 114–129. doi:10.1504/IJICS.2017.082842.
[139] H. Li, K. Gai, L. Zhu, P. Jiang, M. Qiu, Reputation-Based Trustworthy Supply Chain Management Using Smart Contract, in: Proceedings of the 20th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP '20), Vol. 12454, Springer, 2020, pp. 35–49. doi:10.1007/978-3-030-60248-2_3.
[140] D. M. S. Velandia, N. Kaur, W. G. Whittow, P. P. Conway, A. A. West, Towards industrial internet of things: Crankshaft monitoring, traceability and tracking using RFID, Robotics and Computer-Integrated Manufacturing 41 (2016) 66–77. doi:10.1016/j.rcim.2016.02.004.
[141] S. Pollard, G. Adams, F. Azhar, F. Dickin, Authentication of 3D Printed Parts using 3D Physical Signatures, in: Proceedings of the NIP & Digital Fabrication Conference, Printing for Fabrication 2018, Society for Imaging Science and Technology, 2018, pp. 196–201. doi:10.2352/ISSN.2169-4451.2018.34.196.

Appendix A. Pseudo Code of our used Supply Chain Traversal Algorithm for Tracking and Tracing

```
1  def trace(traceid):
2      queue = Queue()
3      records = Map()
4      queue.put(traceid)
5      while not queue.empty():
6          recordId = queue.get()
7          if recordId in records:
8              continue
9          record = API.getRecord(recordId)
10         records.add(recordId, decrypt(record))
11         for reference in record.tracing_references:
12             queue.put(decrypt(reference))
13     return records
```

Listing 1: Pseudo code of the algorithm used for collaborator-based tracing in PrivAccIChain.The utilization of a queue enables multiple threads or processes to request records in parallel.The algorithm further keeps track of already requested records to avoid duplicated requests and non-terminating reference loops.However, the latter can by design (cf. Section 2.1) only occur upon malicious construction of other collaborators.