future internet

MDPI

Article

TrustShare: Secure and Trusted Blockchain Framework for Threat Intelligence Sharing

Hisham Ali [1,*], William J. Buchanan [1,*], Jawad Ahmad [2], Marwan Abubakar [1], Muhammad Shahbaz Khan [1] and Isam Wadhaj [1]

[1] Blockpass ID Lab, Edinburgh Napier University, Edinburgh EH10 5DT, UK; m.abubakar@napier.ac.uk (M.A.); muhammadshahbaz.khan@napier.ac.uk (M.S.K.); i.wadhaj@napier.ac.uk (I.W.)

[2] Cybersecurity Center, Prince Mohammad Bin Fahd University, Al Khobar 31952, Saudi Arabia

* Correspondence: h.ali@napier.ac.uk (H.A.); b.buchanan@napier.ac.uk (W.J.B.)

Abstract

We introduce TrustShare, a novel blockchain-based framework designed to enable secure, privacy-preserving, and trust-aware cyber threat intelligence (CTI) sharing across organizational boundaries. Leveraging *Hyperledger Fabric*, the architecture supports fine-grained access control and immutability through smart contract-enforced trust policies. The system combines *Ciphertext-Policy Attribute-Based Encryption (CP-ABE)* with temporal, spatial, and controlled revelation constraints to grant data owners precise control over shared intelligence. To ensure scalable decentralized storage, encrypted CTI is distributed via the *IPFS*, with blockchain-anchored references ensuring verifiability and traceability. Using *STIX* for structuring and *TAXII* for exchange, the framework complies with the *GDPR* requirements, embedding revocation and the right to be forgotten through certificate authorities. The experimental validation demonstrates that TrustShare achieves low-latency retrieval, efficient encryption performance, and robust scalability in containerized deployments. By unifying decentralized technologies with cryptographic enforcement and regulatory compliance, TrustShare sets a foundation for the next generation of sovereign and trustworthy threat intelligence collaboration.

Keywords: distributed ledger technology; cyber threat intelligence; encryption; GDPR

check for updates

Academic Editor: Demetris Trihinas

Received: 27 April 2025
Revised: 23 June 2025
Accepted: 24 June 2025
Published: 27 June 2025

Citation: Ali, H.; Buchanan, W.J.; Ahmad, J.; Abubakar, M.; Khan, M.S.; Wadhaj, I. TrustShare: Secure and Trusted Blockchain Framework for Threat Intelligence Sharing. *Future Internet* **2025**, *17*, 289. https://doi.org/10.3390/fi17070289

1. Introduction

Cyber threat intelligence (CTI) sharing is essential for proactive defense against evolving cyber threats. However, the existing systems face challenges in ensuring trust, security, privacy, automation, and regulatory compliance. This paper addresses these challenges by proposing TrustShare, a decentralized trust-aware infrastructure that enables secure and efficient threat intelligence sharing across diverse stakeholders, as shown in Figure 1.

We begin by analyzing the limitations of the current solutions through a systematic review of the literature, expert interviews, and industry feedback. These investigations revealed critical gaps in data integrity, trust evaluation, and compliance with frameworks like the General Data Protection Regulation (GDPR)—particularly the *right to be forgotten (RtbF)*.

To address these gaps, we designed a framework that integrates the following technologies and standards: *Hyperledger Fabric* for permissioned blockchain support, *Structured Threat Information eXpression (STIX)* and *Trusted Automated Exchange of Intelligence Information (TAXII)* for standardized threat exchange, *Ciphertext-Policy Attribute-Based Encryption (CP-ABE)* for access control, and the *Interplanetary File System (IPFS)* for scalable off-chain data storage. The framework is further aligned with the *MITRE ATT&CK* matrix for comprehensive threat modeling.

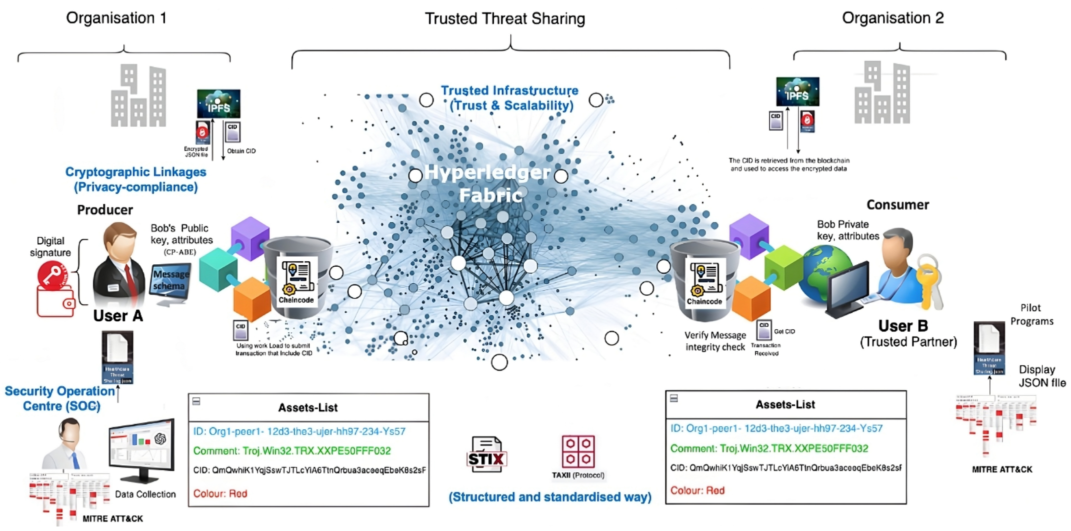
Organisation 1
Trusted Threat Sharing
Organisation 2
IPFS
Encrypted JSON file
CID
Obtain CID
Cryptographic Linkages
(Privacy-compliance)
Producer
Digital signature
Bob's Public key, attributes
(CP-ABE)
Message schema
User A
Chaincode
CID
Using work Load to submit transaction that include CID
Security Operation Centre (SOC)
Data Collection
MITRE ATT&CK
Trusted Infrastructure
(Trust & Scalability)
Hyperledger Fabric
Assets-List
ID: Org1-peer1- 12d3-the3-ujer-hh97-234-Ys57
Comment: Troj.Win32.TRX.XXPE50FFF032
CID: QmQwhiK1YqjSswTJTLcYlA6TtnQrbua3aceeqEbeK8s2sF
Colour: Red
STIX
TAXII (Protocol)
(Structured and standardised way)
The CID is retrieved from the blockchain and used to access the encrypted data
Consumer
Bob Private key, attributes
Chaincode
Verify Message integrity check
CID
Get CID
Transaction Received
User B
(Trusted Partner)
Pilot Programs
Display JSON file
Assets-List
ID: Org1-peer1- 12d3-the3-ujer-hh97-234-Ys57
Comment: Troj.Win32.TRX.XXPE50FFF032
CID: QmQwhiK1YqjSswTJTLcYlA6TtnQrbua3aceeqEbeK8s2sF
Colour: Red
MITRE ATT&CK

Figure 1. TrustShare workflow for secure threat intelligence sharing.

1.1. Problem Statement and Motivation

Cyber threat intelligence (CTI) sharing entails the exchange of sensitive and contextual security information among organizations that often operate in mutually distrusting environments. While centralized CTI sharing architectures are widely used, they exhibit fundamental limitations related to trust, auditability, compliance, and resilience:

- **Trust and Autonomy:** Centralized systems require participants to trust a single managing authority. This is problematic in cross-sector scenarios (e.g., finance and healthcare), where entities prioritize operational independence and data sovereignty.
- **Auditability and Tamper Resistance:** Traditional platforms frequently lack verifiable audit trails and tamper-resistant logs. In security-sensitive domains, such capabilities are essential for traceability and forensic analysis.
- **Regulatory Compliance:** Legal frameworks such as the General Data Protection Regulation (GDPR) impose requirements including the right to be forgotten (RtbF), fine-grained access control, and revocation capabilities. Centralized solutions often struggle to provide transparency and enforcement guarantees in this context.
- **Availability and Resilience:** Single points of failure in centralized systems expose them to outages and targeted attacks. Decentralized alternatives offer improved fault tolerance and continuous availability.

While private blockchains offer control and efficiency, they sacrifice decentralization, introduce governance and insider risks, and face challenges like limited transparency and vendor dependency. These limitations highlight the need for a decentralized architecture that supports secure, policy-compliant CTI sharing.

In this paper, we explore the design of such an architecture guided by the following key questions:

- How can CTI be shared securely in a decentralized trustless environment while ensuring regulatory compliance (e.g., the GDPR)?
- What cryptographic access control mechanisms are best suited for attribute-based, revocable, and selective data disclosure?
- Can decentralized platforms based on Hyperledger Fabric and the IPFS meet the latency, scalability, and interoperability demands of operational CTI workflows?

To address these challenges, we propose **TrustShare**—a decentralized CTI sharing framework built on permissioned blockchain (Hyperledger Fabric) and equipped with the following capabilities:

- Fine-grained access control using Ciphertext-Policy Attribute-Based Encryption (CP-ABE).
- Encrypted off-chain data storage via the Interplanetary File System (IPFS).
- Smart contract-based policy enforcement.
- Regulatory compliance features supporting the RtbF and attribute-based revocation.

While decentralized solutions may introduce performance overheads compared to their centralized counterparts, the resulting gains in transparency, trust, compliance, and operational resilience justify their adoption in inter-organizational CTI sharing ecosystems.

1.2. Contributions

The key contributions of this paper are as follows:

- **Design of TrustShare:** A novel regulation-compliant data sharing framework built on Hyperledger Fabric, supporting secure, privacy-preserving, and policy-driven cyber threat intelligence (CTI) exchange among mutually distrusting entities.
- **Fine-Grained Cryptographic Access Control:** Integration of Ciphertext-Policy Attribute-Based Encryption (CP-ABE) with advanced constraints, such as temporal validity, controlled revelation, and geolocation interlocking—enabling dynamic and sovereign data access control.
- **Hybrid Trust Enforcement via Smart Contracts:** Implementation of a programmable trust-aware access control logic using Hyperledger Fabric chaincode, combining direct (interaction-based) and indirect (reputation-based) trust metrics to automate sharing decisions.
- **Decentralized Storage Integration:** Coupling the IPFS with blockchain-anchored identifiers and CP-ABE-encrypted data to achieve scalable, efficient, and tamper-proof off-chain storage with verifiable access policies.
- **Compliance with Privacy Regulations:** Embedding GDPR-aligned features—including data minimization, revocation via certificate authorities, and auditability—into the design, ensuring the enforceability of the right to be forgotten (RtbF) and lawful data processing.
- **Practical Evaluation:** Demonstration of low-latency, high-throughput, and resource-efficient performance across distributed deployment environments, highlighting TrustShare's scalability and operational viability for real-world CTI alliances.

To clarify our scientific contribution, we frame this work as the first to integrate a hybrid trust model (combining interaction-based and reputation-based scoring) with fine-grained access control (CP-ABE) and GDPR-compliant mechanisms in a blockchain-based CTI sharing platform. Our contributions go beyond assembling technologies: we introduce a modular framework that supports sovereign, secure, and audit-ready CTI collaboration across distrusting organizational boundaries and various technological solutions. The system is validated through a full-stack prototype with performance evaluations under realistic conditions.

Our proposed infrastructure is particularly applicable to Security Operation Centers (SOCs) in high-risk sectors such as healthcare, finance, and government. It supports collaborative defense by enabling trusted intelligence exchange while adhering to modern regulatory and operational demands.

2. Background

This section outlines the foundational components of threat intelligence systems, with a focus on two core functions: *preparation* and *sharing* of cyber threat intelligence (CTI). We examine how structured data formats, encryption, and distributed architectures contribute to effective threat information management. Additionally, we provide a comparative analysis of the related works in the field, highlighting their limitations and illustrating how our proposed TrustShare framework addresses the key gaps in trust, privacy, automation, and regulatory compliance.

2.1. Cyber Threat Intelligence Overview

While conventional security tools such as antivirus software and endpoint detection and response solutions exhibit a degree of effectiveness, their capability falters in identifying emerging threats like zero-day attacks [1]. The escalating threat landscape emphasizes the growing significance of cyber threat intelligence, providing insightful analysis on both potential and existing threats to generate valuable intelligence information. Sharing this information elevates defense systems from passive and active to a proactive approach through collaborative efforts [2].

Cyber threat intelligence refers to the collection, analysis, and production of structured data and intelligence information about potential cyber threats and then disseminating it among trusted organizations. Threat intelligence encompasses insights into potential cyber threats and risks to an organization's information systems, infrastructure, and data [3].

CTI involves gathering information from various sources to convert raw data into information and then into actionable intelligence. These sources often include internal logs, external threat feeds, open-source intelligence (OSINT), dark-web monitoring, security incidents, and event management systems [4,5].

Data gathered by cyber threat intelligence is analyzed to identify patterns, trends, and possible risks. Following that, this intelligence is disseminated among security communities in order to enhance collaborative activities intended to improve overall security.

Figure 2 illustrates the CTI lifecycle stages. The threat intelligence lifecycle involves

1. Data collection from different sources.
2. Processing and analysis.
3. CTI production.
4. Dissemination among trusted users and a feedback loop.

Effective CTI, which requires the expertise of skilled analysts, the utilization of advanced technologies, and a strong emphasis on collaboration, enables organizations to proactively identify and mitigate risks, thereby enhancing their overall security postures [3].

The purpose of cyber threat intelligence is to enable organizations to make informed decisions and take proactive measures to prevent and mitigate cyber attacks. It plays a crucial role in cybersecurity by providing valuable insights into emerging threats, helping organizations to stay one step ahead of adversaries [6–8].

CTI is categorized into different levels, including actionable intelligence, tactical intelligence, and strategic intelligence, each serving distinct purposes, as illustrated in Table 1 [2,9].

The growing capabilities of CTI platforms play a significant role in enhancing information consumption and analysis. This, in turn, facilitates the seamless sharing of threat information within the cybersecurity community [10]. Unstructured text processing, particularly in incident-related information from OSINT and cybersecurity forums, is crucial for identifying tactics, techniques, and procedures (TTPs) and detecting dark web forums or sources where attacks are explained [11]. This holistic approach allows for the effective detection of attacks, understanding shared information networks, and identifying associated threat actor groups.

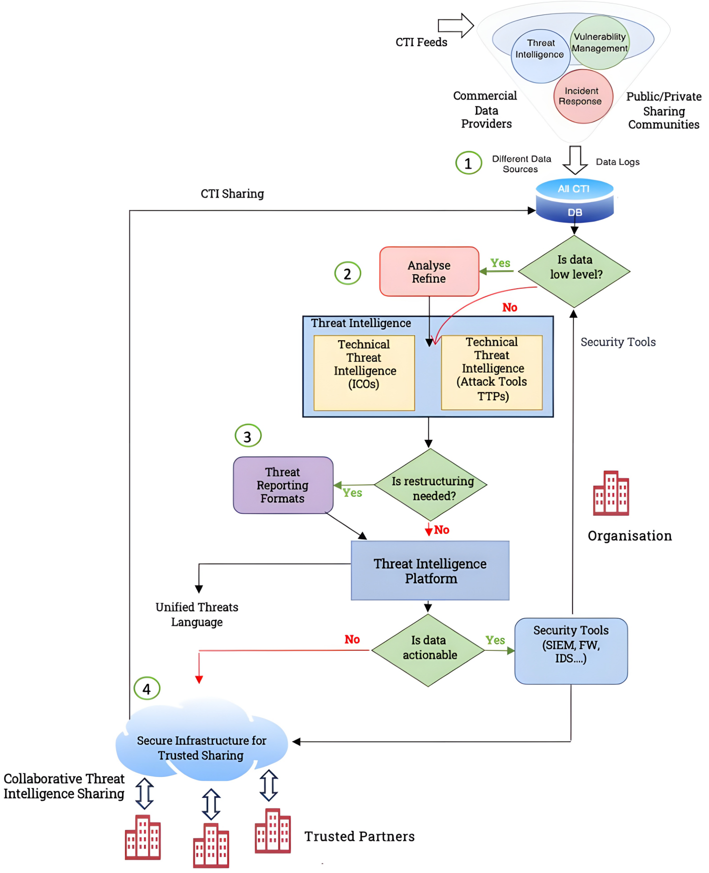

Figure 2. Threat intelligence lifecycle involves data collection, analysis, and dissemination among trusted partners.

Table 1. Cyber threat intelligence categories.

Category	Description	Focus	Example
Actionable Intelligence	Specific, relevant, and timely information for immediate threat response.	Directly applicable details for enhancing cybersecurity defenses.	Indicators of compromise (IoCs); specific vulnerabilities.
Tactical Intelligence	In-depth insights into the techniques and procedures used by threat actors.	Understanding methods to adapt and refine security measures.	Analysis of attack patterns; tactics employed by threat actors.
Strategic Intelligence	Broader long-term perspectives on threat actors' goals and motivations.	Assists in shaping the overall cybersecurity strategy by predicting future threats.	Industry-specific threat landscapes; geopolitical influences on cyber threats.

In summary, cyber threat intelligence is of utmost importance in today's rapidly evolving cyber threat landscape. CTI sharing empowers organizations to proactively protect their assets by providing actionable insights and helping in risk assessment, incident response, and proactive defense strategies.

2.2. Blockchain Overview

Blockchain, a unique and promising technology, has brought tangible changes to the cybersecurity field. The first appearance of blockchain was in 2008 regarding Nakamoto [12,13], which was associated with the Bitcoin cryptocurrency and the digital financial sector. Fundamentally, blockchain was innovated to secure Bitcoin transactions and ensure transaction is received by the concerned person.

Lately, it has demonstrated its validity, efficacy, and efficiency in various areas. As a result, it became extensively used in information systems and computer network security with its different applications.

Blockchain is a decentralized digital ledger that stores electronic records [14]. It basically works as a network of computers, or nodes, scattered among several devices, each of which maintains transactions and data. Blockchain relies heavily on nodes, which represent computers. Transactions undertaken during a certain period are organized into blocks, each with its own unique identification known as a hash. Each block includes the hash of the previous block [15]. A hash is the result of a cryptographic procedure that generates a fixed-size output from any amount of input data. The aforementioned procedure is irreversible.

Transaction authentication in the blockchain includes cryptographic keys (public and private keys). When a transaction is requested, a block with the transaction is created and broadcast to the whole network. Before authorizing a transaction and adding a block to the chain, the network nodes must reach a consensus. The consensus method secures transaction acceptance by requiring a majority of nodes to verify the transaction. Notably, nodes participate in mining, which is the process of solving complicated mathematical problems (algorithms) in order to verify transactions. This procedure, known colloquially as proof of work, emphasizes the decentralized and secure character of blockchain networks.

Currently, PoW is the most widely used mechanism for mining. However, PoW requires substantial computing power and therefore uses considerable amounts of energy, a notable drawback. To solve this issue, another mining mechanism, proof of stake (PoS), is becoming popular. PoS provides faster transactions and uses less energy during mining [16].

Consequently, blockchain provides a secure and trusted method as it has gained a good reputation among existing security techniques. There are key features that make blockchain have great potential to be valid for various security applications, such as immutability and transparency, traceability, and providing a trusted method whereby verification and validation are conducted without the need for trusted signers or third parties. Some significant properties of blockchain are outlined as follows [17]:

- **Decentralized:** Decentralized systems eliminate the requirement of a central authority to validate transactions, instead relying on a consensus algorithm. This approach mitigates the risk of a single point of failure in the blockchain system. Each node in the network maintains a complete and verified record of transactions, ensuring a high level of data redundancy and availability.
- **Immutability:** Once a block is added to the chain, data becomes nearly immutable, ensuring security. Blockchain uniquely allows data ownership to be changed only by the owner, and its origin is traceable, serving as a reliable ledger. The use of the National Institute of Standards and Technology (NIST)-certified cryptographic methods, like SHA-256 and 256-bit ECDSA, enhances user identity security and preserves digital assets effectively.
- **Anonymity:** Anonymity within the context of blockchain technology grants participating nodes the capability to engage in transactions and activities without the necessity of revealing their individual identities. This feature allows users to contribute and interact within the blockchain network while maintaining a level of privacy and confidentiality regarding their personal information.
- **Auditability:** At any point in time, the blockchain allows for the validation of existing transactions. This involves verifying that a transaction has not been altered over time by examining the cryptographic hashes associated with the preceding blocks. This process ensures the integrity and security of the transaction history recorded on the blockchain.
- **Trustless and Transparency:** Nodes can transact without pre-established trust, and all transactions are recorded on the ledger to ensure transparency. Each transaction is recorded on the blockchain, making it visible to every node in the network.
- **Use of Smart Contracts:** Smart contracts enable the automation of transactions within the blockchain. The computer code enhances computational efficiencies by facilitating and verifying agreements among nodes.

Moreover, the blockchain consensus algorithm operates at a slower pace because nodes need time to agree on adding a new transaction block to the chain. This process is more time-consuming compared to the conventional transaction mechanism in centralized systems. While blockchain provides valuable features, it faces challenges in scalability, censorship, latency, and privacy. To address these concerns, integrating blockchain with compatible technologies like the IPFS and implementing strong security measures, such as encryption techniques, delegated credentials, trusted signing, and smart contracts, can be effective.

Blockchain ensures transparent and secure transactions and unmodifiable content by allowing any node in the decentralized network to observe transactions. However, its public nature raises privacy concerns [18]. In contrast, privacy is a focus in other types, using smart contracts to govern rules on specific nodes, as seen in Ethereum and permissioned blockchain platforms. Blockchain types vary based on user authorization and authentication, ranging from private and consortium blockchains to public blockchains. See Table 2 for features of different blockchain classifications.

Table 2. Comparison of different blockchain types.

Comparison Features	**Public Blockchain (Permissionless)**	**Private Blockchain**	**Permissioned Blockchain**
Read	It is an open network, no permission needed, anyone can download the protocol and read	Only specific participants in the organization can read, verify and add new nodes	Under a legal contract, the public and participants are permissible
Write	It is an open network, no permission needed, anyone can download protocol and write	Only specific participants in the organization can write	Participants are permissible under some legal contracts
Consensus Operational	No conditions are needed to join consensus; the process needs more energy and resources.	Only those who are pre-selected can conduct the consensus within the organization	Pre-selected nodes within consortium
Examples	**Bitcoin**	**Ethereum**	**Hyperledger Fabric**
Network	Permissionless	Permissioned or Permissionless	Permissioned
Classification	Public	Public or Private	Private
Governance	Decentralized	Ethereum Developers	Linux Foundation
Currency	Yes	Yes, Ether Tokens (smart contract)	None, Currency Tokens (chaincode)
Operation Pattern	Order–execute	Order–execute	Execute–order–validate
Cost	Yes, Satoshi (It is synonymous with Bitcoin).	Yes, Gas (the amount of computational power).	None
Smart Contracts	No	Smart Contracts (Solidity, Serpent, and LLL)	Chaincodes (Go, JavaScript, Java, and more)
Consensus Algorithms	Proof of Work	Proof of Work or Proof of Stake (new versions)	Normal operation or Practical Byzantine Fault Tolerance (PBFT)
Encryption of transaction data	No	No	Yes
TPS	3.3–4.6	15	Up to 5000
Block Size	1–2 MBs 4 MB SegWit (Segregated Witness)	20–30 KB	By default: 512 KB (Preferred) 98 MB (Absolute Maximum)
Transactions per Block	3500	70	10 (default)
Block Time	10 min	15 s	1 s
Currency Capitalization	21 million	5 every 14 s	No Currency
Current Block Reward	12.5 BTC	3 ETH	N/A
Applications	Track ownership of Digital Currency (Mostly)	DApps (Games, IoT, Fintech, Supply Chain, and so on)	Private Blockchain requires High Performance, resiliency, and privacy

2.3. Private Blockchain

Historically, threat information sharing has relied on manual modeling and centralized network systems, which can be inefficient, insecure, and prone to errors. Alternatively, private blockchains are now widely used to address these issues and improve overall organizational security.

In the context of sharing threat intelligence, we opt for a blockchain solution instead of relying on a centralized trusted third party. The participating parties in information sharing aim to mitigate breaches by adversaries while acknowledging mutual distrust. In the process of choosing a blockchain solution, our methodology aligns with the framework proposed in [19], with some more features to assess its eligibility for threat intelligence sharing using private blockchains. To simplify and support the decision-making process, Table 3 compares different private blockchain choices.

Table 3. Comparison of private ledgers.

Feature	Hyperledger Fabric	Ethereum	Quorum	R3 Corda
Consensus Mechanism	Practical Byzantine Fault Tolerance (PBFT)	Proof of Stake (PoS)	Istanbul BFT (IBFT)	RAFT consensus
Smart Contracts	Chaincode (Go and Java)	Solidity	Smart Contracts (Solidity and Vyper)	Contracts (Kotlin and Java)
Privacy	Channel-based privacy	Limited privacy features	Private transactions with Constellation	Corda Firewall and transaction privacy
Token/Currency	Customizable (No built-in currency)	Ether (ETH)	Quorum Token (Quorum)	Customisable (No built-in currency)
Permissioning	Granular control with Membership Services	Public and private networks	Permissions for nodes and transactions	Access Control in Corda Network
Performance and Scalability	High scalability with modular architecture	Scalability challenges, higher latency	Scalability improvements, moderate latency	Designed for scalability and performance
Development Community	Active development community	Large and an active community	Developed by JPMorgan, active community	Active community with enterprise focus

In order to evaluate the feasibility of using a private blockchain to share threat information in a way that protects user data privacy, we created this table. Additionally, encrypting data and storing it off-chain is proposed using an external data store. To implement threat information sharing, at least two parties must interact through a smart contract, ensuring private data is not redistributed to unauthorized parties. Participants in conventional systems typically provide identification to a reliable third party, which may not always be online. Members know each other yet do not trust each other. Finally, public verifiability is not desirable since readers should also be limited. Consequently, a private permissioned blockchain is appropriate for our use case, as we have compared different blockchain criteria.

Research was conducted on open-source options for our use case. Hyperledger Fabric provides private blockchain solutions [20], as do R3 Corda [21] and Quorum [22]. Ethereum [23] may also be set to run in private mode; therefore, it can be added to this list. We conducted an analysis of these four options in order to select the one that aligns with the research objectives. Unlike Hyperledger Fabric and Ethereum, Quorum and R3 Corda were designed with the financial sector over the years [24,25]. Another key feature is the ability to write smart contracts in a programming language, simplifying their creation. In Hyperledger Fabric, Quorum, and R3 Corda, distributed smart contracts may be authored in general-purpose programming languages. However, Ethereum only supports smart contracts written in Solidity, a domain-specific language. Private blockchain solutions do not necessitate the use of pricey PoW technologies as the agreement mechanics. As a result, each solution offers possibilities with varying levels of robustness. Hyperledger Fabric uses the Practical Byzantine Fault Tolerance (PBFT) consensus technique and also supports BFT-based solutions. Quorum's consensus protocols include Istanbul BFT (IBFT). R3 Corda also supports consensus methods based on RAFT consensus.

Private Ethereum blockchains, on the other hand, can use the proof of stake (PoS) protocol. Notably, unlike Ethereum, other blockchain options do not require native coins for intelligent functioning. Ethereum, on the other hand, requires Ether to charge computational fees. We did not select Quorum or Corda because they are mostly used for financial applications. Hyperledger Fabric has less community and research support than Ethereum. Ethereum, on the other hand, has less throughput and more delay than Hyperledger Fabric [25,26]. There are additional crucial considerations when selecting Hyperledger Fabric as any currency is unacceptable. On top of that, general-purpose languages are supported. Last but not least, many studies are working to improve Hyperledger Fabric's security, performance, and reliability issues with scalability [27–29]. Several features of the private blockchain solutions covered in [25] are compared, and the results of the research analysis can be seen in Table 3.

Hyperledger Fabric stands as the cornerstone of a private permissioned blockchain framework meticulously chosen to craft a proof of concept (PoC) for threat information sharing. Through this option, we aim to optimize the seamless exchange of threat information among trusted organizations. In enhancing threat intelligence sharing, organizations must prioritize secure communication of threat data through encrypted or private channels. This approach ensures confidentiality and leverages trusted infrastructure to automate threat sharing, ultimately reducing threat detection and incident response times. Hyperledger Fabric plays a vital role in this context by emphasizing rigorous identity verification for all participating entities [20].

Utilizing chaincode, similar to Ethereum's smart contracts and writable in Java, JavaScript, or Go, a consortium of companies can seamlessly coordinate their own blockchains within the framework [20]. Hyperledger Fabric's intentional design emphasizes modularity, scalability, and versatility, forming a robust foundation for privacy, confidentiality, and scalability. In contrast to conventional blockchain approaches, its operational phases adhere to the "execute–order–validate" pattern, a strategic departure that enhances adaptability and effectiveness, aligning with trustworthy threat intelligence sharing in this context. This ensures transactional processes supported by consensus and fault-tolerant algorithms [20].

Administrative duties in blockchain include setting up consensus protocols and replacing resource-intensive algorithms like proof of work or proof of stake with more efficient alternatives, such as Paxos, RAFT, or BFT, in order to enhance blockchain performance [30]. Transactions encompass multiple stages, such as chaincode execution, identity verification, orderer authorization, and peer ledger modifications [20]. APIs and SDKs, specifically Node.js and Java SDKs, are utilized for orchestration, as stated in the official documentation [31].

Entities in Hyperledger Fabric take the form of Docker containers, seamlessly integrated with the host operating system [32]. Scalability is a key focus, addressing peers, organizations, ordering services, and channels [32]. Peers play a vital role, serving dual functions by hosting ledgers and chaincodes and enabling private interactions through channels within their organizations [33]. Notably, utilizing CouchDB enables more efficient query execution compared to other relational databases, leading to reduced latency, especially with simpler queries [33].

Ordering services responsible for ledger validation and updating follow consensus protocols like Paxos or RAFT [20]. The gossip protocol, as outlined in [20], facilitates system resilience by ensuring continuous peer communication and updates, even in the face of disruptions. Privacy enhancements, rolled out in version 1.2, now enable authorized participants to have exclusive access to private data collections [33].

The resilient security framework of Hyperledger Fabric is emphasized through advanced features like private data collections and zero-knowledge proof using Identity Mixer (Idemix) [34]. Participants within the system leverage X.509 digital certificates issued by a certificate authority (CA) and authenticated by a membership service provider (MSP) [20]. Beyond the core framework, various projects like Hyperledger Iroha, Burrow, Cello, Composer, and Explorer contribute to the continuous expansion of the Hyperledger family, enriching its overall capabilities [30].

To contextualize our proposed trust framework, Table 4 presents a comparative analysis of the existing blockchain-based threat intelligence sharing systems. The comparison focuses on the employed trust models, blockchain platforms, implementation status, scalability, and unique contributions. While previous works such as those by Homan et al. [35] and Zhang and Miao [36] offer foundational trust and sharing mechanisms, they often lack real-world scalability or do not integrate advanced access control. In contrast, our proposed system, TrustShare, introduces a hybrid trust model combining direct and indirect evaluations, leverages Hyperledger Fabric's modular architecture for enhanced scalability, and incorporates CP-ABE for fine-grained access control. This comparative perspective highlights the novelty and practicality of our approach in addressing the current limitations in secure and privacy-preserving threat intelligence sharing.

Table 4. Comparison of trust models in blockchain-based threat intelligence sharing systems.

Work	Trust Model	Blockchain Platform	Real Implementation	Scalability	Unique Feature
Homan et al. (2019) [35]	No explicit trust model; assumes honest participants	Hyperledger Fabric	Testbed implementation with STIX 2.0	Moderate; evaluated with Hyperledger Fabric channels	Focus on GDPR compliance and data segmentation using Hyperledger Fabric channels
Zhang and Miao (2021) [36]	Reputation-based trust model	Consortium Blockchain	Simulation-based evaluation	Not specified; focuses on security and trust	Introduces proof of reputation consensus algorithm for CTI sharing
Nguyen et al. (2021) [37]	Incentive-based trust model combining monetary and reputation incentives	Hyperledger Fabric	Prototype implementation with IPFS integration	Moderate; utilizes Hyperledger Fabric's modular architecture	Incentivized CTI sharing framework tailored for Industrial Control Systems (ICSs)
Ma et al. (2023) [38]	Evolutionary game theory-based incentive mechanism	Ethereum	Simulation and smart contract deployment	Not specified; focuses on incentive mechanism effectiveness	Addresses free-riding behavior in CTI sharing using game-theoretic approach
Hu et al. (2024) [39]	Trust management integrated in multi-blockchain architecture	Multi-Blockchain (BCH, ETH, and LTC)	Prototype implemented for secure medical data sharing	High; supports distributed data storage across chains	Proposes a hierarchical model for multichain integration with fine-grained access control
TrustShare (This Work)	Hybrid (Direct + Indirect)	Hyperledger Fabric (Private)	Full-stack implementation with smart contracts	High scalability via modular and channel-based architecture	CP-ABE for access control, GDPR compliance, and STIX/TAXII integration

2.4. Comparative Analysis and Innovation Highlights

The field of cyber threat intelligence (CTI) sharing has witnessed significant advances in recent years, yet several limitations persist in the existing frameworks. A comparative review of the related work reveals key challenges that have guided the design of the TrustShare framework:

- **Limited Support for Fine-Grained Access Control:** Many existing blockchain-based CTI systems lack attribute-based access policies, instead relying on static role-based models. This restricts flexibility in cross-organizational sharing and fails to meet the compliance needs for highly regulated sectors.
- **Inadequate Alignment with the GDPR and Data Sovereignty Requirements:** The ability to enforce consent-driven auditable access to sensitive data is often missing in prior work. Systems without privacy-preserving encryption or explicit access constraints risk breaching regulatory obligations.
- **Absence of Integrated Trust Models for Data Providers and Consumers:** Most platforms assume all nodes are equally trusted or use binary trust evaluations. This oversimplifies real-world collaboration, where entities have varying reputations and behavior patterns.

In response, **TrustShare introduces the following innovations**:

- **CP-ABE-Based Fine-Grained Encryption:** By integrating Ciphertext-Policy Attribute-Based Encryption (CP-ABE), TrustShare allows data owners to define expressive access control policies directly embedded in the encrypted payloads, ensuring only authorized recipients can decrypt shared intelligence.
- **Hybrid Trust Scoring Model:** We propose a novel combination of direct and indirect trust metrics to assess participant reliability over time. Trust scores influence access and endorsement privileges, thereby promoting collaboration only among reputable parties.
- **Smart Contract-Enforced Policy Compliance:** Smart contracts codify organizational and regulatory policies (e.g., minimum trust thresholds and data handling restrictions), enabling automated verifiable enforcement and audit logging on the blockchain.

Table 5 summarizes the differences between TrustShare and the existing frameworks across key functional dimensions. These innovations collectively enable secure, compliant, and trust-aware CTI sharing suitable for deployment in multi-stakeholder environments such as national CERTs, healthcare consortia, and financial threat sharing alliances.

Table 5. Summary comparison of TrustShare with existing CTI sharing frameworks.

Framework	Access Control Model	Privacy Compliance	Trust Model	Enforcement Mechanism	Unique Innovation/ Comments
Existing Systems	Role-based or None	Limited or None	Binary/Static	Off-chain or Manual	Limited fine-grained control and weak privacy enforcement
TrustShare (This Work)	CP-ABE (Attribute-Based)	GDPR-aligned, Auditable	Hybrid Dynamic Trust Scores	Smart Contract-Automated	Fine-grained encryption policies, trust-aware access, automated compliance enforcement

3. Materials and Methods

In this section, we propose an innovative framework for secure and efficient threat intelligence sharing, leveraging advanced technologies such as Structured Threat Information eXpression (STIX), Trusted Automated Exchange of Intelligence Information (TAXII), Ciphertext-Policy Attribute-Based Encryption (CP-ABE), the Interplanetary File System (IPFS), the MITRE ATT&CK framework, the GDPR, and the right to be forgotten (RtbF). The methodology integrates these technologies to address the current challenges in cybersecurity, ensuring real-time, trusted, and collaborative information exchange among organizations. The following sections detail the materials used and the step-by-step methodology followed to develop and validate this framework.

3.1. Materials

In building a secure system for sharing threat intelligence within distributed ledger technologies (DLTs), our exploration relies on carefully chosen materials. This section explores key technological components and protocols, including Hyperledger Fabric, STIX, TAXII, CP-ABE encryption protocols, and IPFS. Through a detailed evaluation, we aim to establish the foundation for a trusted and secure ecosystem to facilitate collaborative threat intelligence sharing.

TrustShare integrates Hyperledger Fabric with IPFS and CP-ABE, using Docker for containerization and Kubernetes for orchestration. The most challenging aspect was the coordination between the blockchain chaincode and external encryption key management. Integrating CP-ABE with Hyperledger Fabric's chaincode required adapting policy enforcement for dynamic attributes (e.g., geolocation and time-bound access). We also faced complexity in managing private data collections, certificate authorities, and secure multi-party interactions, which we resolved using modular identity mapping and membership services.

3.1.1. Hyperledger Fabric Components and Features

The investigation conducted in this study forms the basis for identifying the most suitable technologies for the proposed sharing infrastructure within the scope of this research. The main focus of our research relies on Hyperledger Fabric technology. Our study comprehensively assessed the cutting-edge platforms, analyzing their features, capabilities, and appropriateness. To augment and articulate our suggested design, we must be aware of the Hyperledger Fabric features and components (see Figure 3), which are defined as follows:

- Hyperledger Fabric Client
  The Hyperledger Fabric client is the front end of the permissioned ledger network that users interact with. To make the threat intelligence sharing ecosystem work seamlessly, it lets users communicate with each other by sending transactions to the nodes and querying the ledger [40].
- Hyperledger Fabric Membership Service Provider
  The system includes the Hyperledger Fabric membership service provider (MSP) and Hyperledger Fabric channels. The MSP manages identity and access control, ensuring secure participant registration and authentication for a robust threat intelligence sharing ecosystem [41].
- Hyperledger Fabric Certificate Authority
  In addition to the aforementioned components, the Hyperledger Fabric certificate authority is integrated into the permissioned ledger network to ensure secure communication and cryptographic authenticity. The implementation of this measure enhances the overall security of the proposed sharing infrastructure ([31]).
- Smart Contracts
  In Hyperledger Fabric, a smart contract, or "chaincode," is executable code defining transaction rules on the blockchain. It automates and enforces business logic, ensuring consistency and transparency. Deployed to channels, it enables secure and verifiable interactions within the network [42].
- Hyperledger Fabric Channels
  Hyperledger Fabric network segmentation using channels is a critical component of our framework selection [43]. These channels make the system more modular and extendable and provide private and secure communication among specified network parties. This enhancement increases the resilience of our proposed sharing ecosystem by improving threat intelligence confidentiality and targeted sharing [44].

- Hyperledger Fabric Peer Nodes
  To augment our proposed framework, we incorporate Hyperledger Fabric peer nodes. These nodes play a vital role in maintaining the shared ledger, endorsing transactions, and ensuring consensus among participants. The inclusion of peer nodes supports network resilience and contributes to the collaborative and secure exchange of threat intelligence [45].
- Hyperledger Fabric Ordering Service
  Expanding our material spectrum, we integrate the Hyperledger Fabric ordering service. This service is instrumental in coordinating the sequencing of transactions, ensuring a consistent and immutable record of shared threat intelligence [46]. The ordering service enhances the reliability and integrity of our proposed sharing infrastructure.

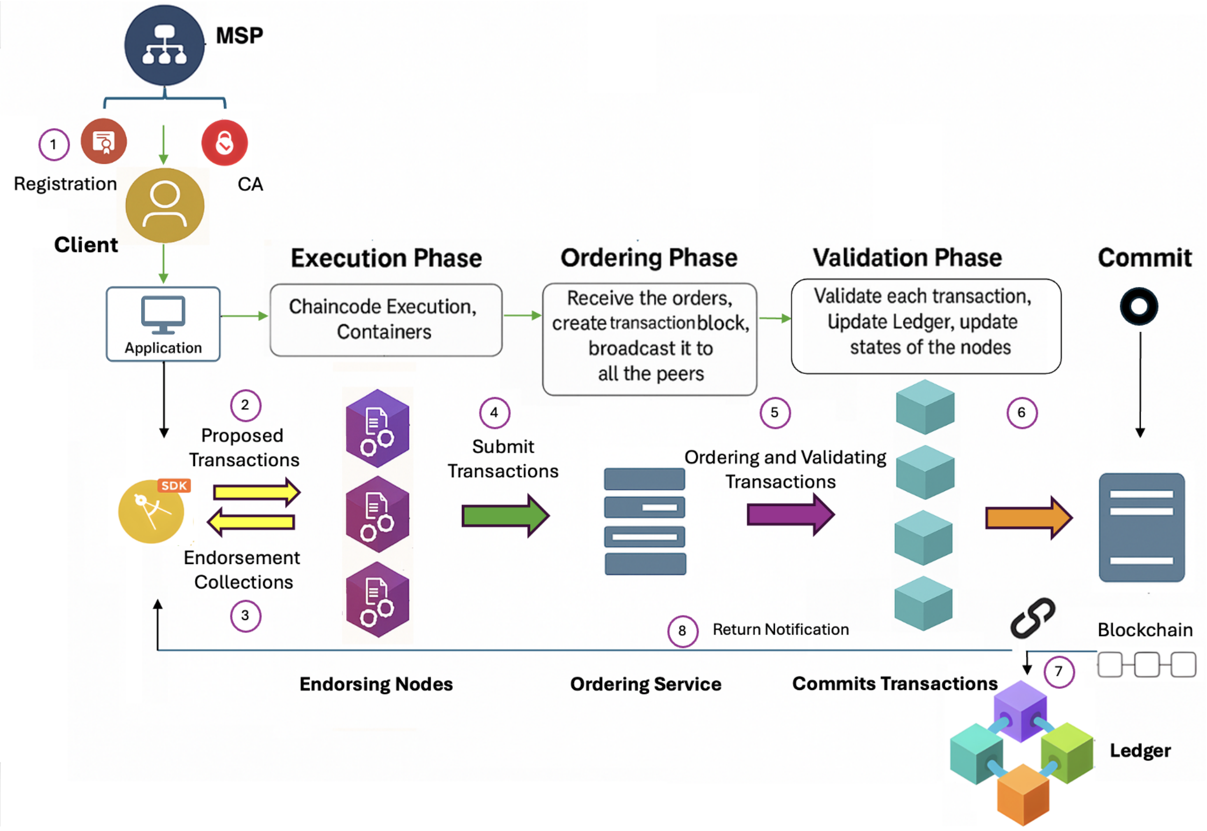

Figure 3. Hyperledger Fabric components and transaction flow.

Our comprehensive evaluation includes an analysis of permissioned ledger technologies, cryptographic protocols, and smart contracts, as well as the integration of key components such as Hyperledger Fabric, channels, the membership service provider, certificate authority, peer nodes, ordering service, and client nodes. The objective of this overview is to establish a robust and secure ecosystem for collaborative threat intelligence sharing.

Ultimately, the optimal choice is based on our requirements and preferences. Therefore, we have chosen Hyperledger Fabric for our project due to its distinct features and objectives, as indicated in the following:

1. *Permissioned blockchain:* Unlike a public blockchain, where nobody needs permission to join the network, Hyperledger Fabric requires an identity and certificate authority for any user or node to join the network [33].
2. *Privacy and confidentiality of transactions:* Channels allow a subset of nodes, through the anchor node, to link different organizations that compose the consortium. The ledger of a channel can be accessed only by those organizations that are part of the

channel. Therefore, participants can view only the network features and data relevant to their specific channel.

3. *Highly modular and configurable architecture:* Hyperledger Fabric enables plug-and-play ordering, membership, endorsement, and validation services. A pluggable consensus algorithm also improves the platform. Ledger supports LevelDB and CouchDB databases [33].
4. *Efficient data query:* By using CouchDB, it could execute queries more efficiently compared to other relational databases with less latency and with simpler queries.
5. *High transaction throughput performance:* Hyperledger Fabric is scalable. Peer nodes are liberated from ordering (consensus) responsibilities, while transaction execution is independent of ordering and commitment. The division of labor relieves the ordering nodes of transaction execution and ledger maintenance.
6. *Low latency of transaction confirmation:* Hyperledger Fabric is considered the fastest amongst all the permissioned blockchains, generated within only a few organizations, thus contributing to reducing latency. Furthermore, it does not have a mining process like blockchain, and it makes the system fast in verifying and committing transactions [31].
7. *Offering multiple languages to write smart contracts:* One can write smart contracts in different programming languages, such as Java, Go, or Node.js [20,33].
8. *No cryptocurrency:* Unlike a public blockchain and several other technologies, Hyperledger Fabric does not involve any cryptocurrency ([33]).
9. *The ability to monitor the network performance:* Hyperledger Caliper is a network benchmarking tool that measures performance using indicators like transactions per second, latency and throughput utilization of resources, and more [47]. It provides the flexibility to specify and configure parameters for conducting tests on the platform, aiming for optimal results.

In light of the above, Hyperledger Fabric is a prominent blockchain framework that effectively addresses security, scalability, and privacy. The versatility and robustness of this solution are attributed to its Docker container-based architecture, private data collections, and diverse language support for chaincode development. This system's enhanced security features, systematic approach to transactions, and adaptability via APIs and SDKs highlight its potential.

3.1.2. Structured Threat Information eXpression (STIX)

STIX is a standardized language and framework for sharing structured threat intelligence across different organizations and security systems [8,48]. It enables the collection, analysis, and sharing of valuable information about threats, vulnerabilities, and related data [6,49,50]. This exchange of intelligence helps organizations to enhance their situational awareness, improve incident response, and collaborate effectively to mitigate security risks. By facilitating the sharing of threat intelligence, STIX aims to create a more unified and proactive defense against cyber threats [51].

The STIX Architecture offers a robust framework for sharing and expressing security threat intelligence. It comprises three main elements: the foundational STIX Core, which structures threat data; the standardized vocabulary of the STIX Language for consistent communication; and the STIX Cyber Observable Object (CybOX), which captures technical observables of cyber threats [8]. This architecture enables effective sharing and analysis of threat intelligence, improving detection and response to security incidents. Furthermore, it fosters interoperability among security tools, automating the exchange of threat intelligence across the cybersecurity ecosystem.

The STIX Data Model is a structured framework for sharing cybersecurity information in a standardized format. It uses a hierarchical structure with entities like indicators and threat actors, connected through relationships. Each entity has specific properties detailing its attributes. This standardized model ensures consistency and interoperability across security tools, allowing organizations to efficiently share, analyze, and act on threat intelligence for enhanced asset and network protection.

STIX proves effective in cybersecurity through various use cases [52]. Organizations collaborate to share threat information, indicators, and incident details, enhancing collective defense. Integration with security tools enables automated analysis, ensuring quicker detection and response to emerging threats. Large Language Models (LLMs) represent a significant advancement in AI technology, offering powerful tools for communication and threat data analysis. These models are used for understanding context, enhancing security, and ensuring proper threat formatting (STIX–TAXII). We proposed the use of Private LLMs to understand the content of threat sharing and to learn from each agency's specific context, whereby we use the Llama dataset and our own data synthesis to enhance datasets and maintain the privacy of our data. This approach can automate the formatting and protection of shared data, enhancing both efficiency and security.

STIX supports proactive threat hunting, allowing analysts to search for potential threats across diverse sources, empowering teams to identify and counter advanced persistent threats [53]. These use cases underscore STIX's value in promoting information sharing, automated analysis, and proactive defense strategies in the cybersecurity community.

Expanding our arsenal for a robust threat intelligence sharing framework within Hyperledger Fabric, we introduce STIX Threat File Sharing. STIX, the globally recognized language for cyber threat information, joins our ensemble to standardize and streamline threat communication. This integration enhances collaboration, breaking down organizational and technological silos for a more universally understood and shared cybersecurity landscape.

In conclusion, Structured Threat Information eXpression (STIX) offers a standardized framework for representing and sharing cyber threat information. Through a common data model, STIX enables structured and machine-readable exchange of threat intelligence. Key concepts like indicators, threat actors, and TTPs allow for consistent and detailed threat representation, enhancing analysis and defensive measures. STIX is applicable in incident response, threat hunting, and intelligence sharing, supporting the exchange of actionable information. It plays a crucial role in enhancing collective cybersecurity efforts and fostering collaboration against cybercrime and advanced persistent threats.

3.1.3. Trusted Automated Exchange of Intelligence Information (TAXII)

Diversifying our infrastructure, we integrate the Trusted Automated Exchange of Intelligence Information (TAXII). As a standard protocol for the exchange of cyber threat intelligence, TAXII facilitates secure and standardized communication of threat information [54]. Its incorporation enhances interoperability and extends the reach of our threat intelligence sharing ecosystem.

This section provides an overview of Hyperledger Fabric and TAXII and explores the possibilities of leveraging these two powerful technologies for threat information sharing. Hyperledger Fabric is a blockchain framework that allows organizations to build private permissioned blockchain networks to enhance the security, transparency, and efficiency of their transactions. TAXII, on the other hand, is a trusted automated exchange protocol that facilitates the sharing of threat information in a standardized and structured manner [48]. By combining Hyperledger Fabric's secure and scalable infrastructure with TAXII's ability to exchange contextualized threat intelligence, organizations can create a robust framework for sharing threat information among trusted parties. We will delve into the features, capabilities, and potential advantages of using Hyperledger Fabric and TAXII together to enable seamless collaboration and enhance cyber threat detection and response.

Integrating Hyperledger Fabric with TAXII offers several compelling benefits for organizations involved in threat information sharing [31,55]. Firstly, it enhances the security and privacy of sensitive threat data. With Hyperledger Fabric's robust framework and consensus mechanism, combined with TAXII's standardized exchange protocols, organizations can ensure the confidentiality and integrity of their shared information. Additionally, this integration enables real-time updates and improved data accessibility. Hyperledger Fabric's distributed ledger technology allows for the immediate propagation of threat information across multiple participants, ensuring that all stakeholders have access to up-to-date and accurate data. Furthermore, by leveraging the smart contract capabilities of Hyperledger Fabric, organizations can automate the enforcement of rules and policies in the sharing process, reducing the risk of human error or malicious activities. This integration also paves the way for improved collaboration and trust-building among organizations. The transparent and auditable nature of Hyperledger Fabric, combined with TAXII's standardized language for threat information, promotes cooperation, fosters trust, and simplifies data sharing between organizations. By integrating these two powerful technologies, organizations can unlock the full potential of threat information sharing, leading to more effective and timely responses to emerging cyber threats.

3.1.4. MITRE ATT&CK Framework

The **MITRE ATT&CK** framework is an invaluable resource for the structured analysis of cyber threats. It provides a comprehensive knowledge base of adversary tactics, techniques, and procedures (TTPs). In our methodology, we leverage MITRE ATT&CK to classify and map gathered threat data, ensuring consistent standardized threat intelligence across participants.

3.1.5. Integrating CP-ABE with Hyperledger Fabric for Enhanced Security and Efficiency

This section provides an overview of the work on "Integrating CP-ABE with Hyperledger Fabric for Enhanced Security and Efficiency," underscoring the importance of merging these technologies to tackle security and efficiency challenges in contemporary distributed ledger systems.

The integration of Ciphertext-Policy Attribute-Based Encryption (CP-ABE) with Hyperledger Fabric is introduced as a solution to augment data privacy, access control, and overall system security [56,57]. CP-ABE facilitates fine-grained access control based on attributes specified by the data owner, while Hyperledger Fabric offers a secure and scalable blockchain framework. This combination allows for improved security by ensuring that only authorized parties can access and modify data, enhancing efficiency through automated access control management. Despite challenges related to integration complexity and potential performance impacts, effective solutions are proposed, including thorough research, optimization techniques, and robust encryption mechanisms [58]. The integration addresses scalability, interoperability, and security concerns, offering notable advantages in terms of data protection, confidentiality, and system reliability. Integrating CP-ABE with Hyperledger Fabric delivers several benefits, including

- Fine-Grained Access Control: CP-ABE facilitates fine-grained access control by encrypting data based on specific attributes, allowing for complex access policies. Users with corresponding attributes can decrypt and access the data [59,60].

- Flexible Attribute-Based Policies: CP-ABE provides flexibility in defining access policies, with attributes representing characteristics like roles or clearance levels. This grants desirable control over data access [61,62].
- Dynamic Policy Enforcement: Access policies can be dynamically updated without re-encrypting the data as attributes change, aligning well with evolving access requirements over time [63].
- Privacy Preservation: CP-ABE preserves privacy by enabling data owners to define access policies without revealing users' actual attributes, which is crucial in scenarios with sensitive attributes [62].
- Secure Data Sharing: Encrypted data can be shared based on attributes without disclosing content, which is beneficial in collaborative environments where multiple parties need specific data access [64].
- Blockchain Immutability and Auditability: Hyperledger Fabric's blockchain ensures immutability and auditability. CP-ABE integration enhances this by securing data at the attribute level, providing a secure and auditable access control mechanism [65].
- Compliance and Regulatory Requirements: CP-ABE aids in meeting stringent regulatory requirements by offering a robust and flexible access control mechanism [66].
- Enhanced Security: CP-ABE adds an extra layer of security to blockchain-stored data. Unauthorized users, even with access, cannot decrypt data without the necessary attributes, reducing the risk of unauthorized data access [67].

It is crucial to carefully assess specific use cases and requirements before implementing CP-ABE due to the added system complexity. The benefits are particularly valuable in scenarios where fine-grained access control and attribute-based policies are essential [68].

In conclusion, the integration of CP-ABE with Hyperledger Fabric presents a promising approach to elevate data security and efficiency in various applications, with ongoing research expected to lead to innovative solutions and advancements in the field.

3.1.6. Interplanetary File System (IPFS)

Broadening our infrastructure, we integrate the Interplanetary File System (IPFS), a decentralized and distributed file storage system known for its enhanced accessibility and availability of shared threat intelligence data [45,69]. The IPFS's unique architecture combines ideas from distributed hash tables, BitTorrent, and Git to ensure secure and tamper-resistant storage. This one-of-a-kind design greatly improves the overall resilience of our envisioned sharing ecosystem [70]. Operating on a peer-to-peer architecture, the IPFS replaces the traditional client-server model and allows users to store and access files based on content rather than location, utilizing a hash-based addressing system [71,72]. This approach offers increased reliability, censorship resistance, and improved performance by eliminating the need for a central server. The IPFS's benefits include content addressing, distributed network architecture for redundancy, versioning capabilities, peer-to-peer file sharing, and security through content verification, making it a decentralized, secure, and efficient solution for file storage and sharing [73]. In practical terms, the IPFS finds applications in decentralized web hosting, content distribution, decentralized social media platforms, and decentralized file backup, showcasing its potential to revolutionize various industries [74–76].

3.1.7. General Data Protection Regulation (GDPR)

The European Union established the General Data Protection Regulation (GDPR) in May 2018 to protect the privacy of personal data shared among EU members and other entities with which they interact [77]. This regulation applies to organizations worldwide that handle the personal data of EU residents, ensuring that they comply with the GDPR's requirements and obtain prior consent for data processing.

The GDPR sets strict guidelines for data authorization and authentication, covering crucial data sharing and reusing aspects. It grants individuals several important rights, including the right to access their data, the right to modify it, and the right to have it deleted. These rights can be exercised individually or in combination, providing comprehensive control over personal data.

3.1.8. Right to Be Forgotten (RtbF)

The right to be forgotten, a key component of the GDPR, pertains to the deletion of personal data upon request [77,78]. While implementing this right across the entire network is impractical and unfeasible, it underscores the need for effective protocols to ensure content can be deleted when requested. This responsibility typically falls on content owners or their authorized representatives to execute such deletions as necessary.

3.2. Methodology

This section outlines the techniques employed in our research, focusing on the method workflow illustrated in Figure 4. It provides a clear description of the key steps and processes involved. The objective is to highlight the approach adopted in our study and its implementation.

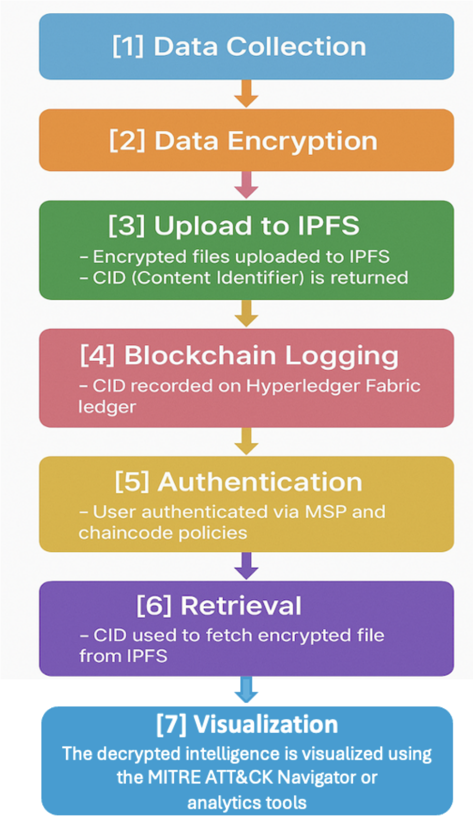
[1] Data Collection
[2] Data Encryption
[3] Upload to IPFS
– Encrypted files uploaded to IPFS
– CID (Content Identifier) is returned
[4] Blockchain Logging
– CID recorded on Hyperledger Fabric ledger
[5] Authentication
– User authenticated via MSP and chaincode policies
[6] Retrieval
– CID used to fetch encrypted file from IPFS
[7] Visualization
The decrypted intelligence is visualized using the MITRE ATT&CK Navigator or analytics tools

Figure 4. Workflow Steps for secure threat intelligence sharing in TrustShare.

Figure 4 presents the sequential steps for securely disseminating threat intelligence in the TrustShare framework, as aligned with the overall process illustrated in Figure 5.

Process Overview: Threat Intelligence Collection and Sharing Workflow in TrustShare

1. **Collect cybersecurity data** from sources such as logs, threat feeds, and private telemetry.
2. **Parse and standardize** the data using the MITRE ATT&CK framework.
3. **Structure the data** as JSON files containing Indicators of Compromise (IOCs).
4. **Encrypt the JSON** using CP-ABE to enforce fine-grained access control.
5. **Upload encrypted JSON** to the Interplanetary File System (IPFS).
6. **Receive the Content Identifier (CID)** from the IPFS.
7. **Submit the CID and user identity** as a transaction to Hyperledger Fabric.
8. **Validate transaction** using Hyperledger Fabric's chaincode.
9. **Immutably store the CID** on the blockchain ledger.
10. **Authorized user requests access** and is authenticated via Hyperledger Fabric's MSP.
11. **Retrieve the CID** from the blockchain ledger.
12. **Use the CID to fetch the file** from the IPFS.
13. **Decrypt the file** using CP-ABE private keys.
14. **Extract threat intelligence** from decrypted content.
15. **Visualize threats** using tools like the MITRE ATT&CK Navigator.

Figure 5. Process overview for secure threat intelligence dissemination using TrustShare (Steps 1–15).

3.2.1. Data Collection and Preprocessing

The threat intelligence sharing process begins with the collection of data from diverse cybersecurity sources. These data sources include open-source feeds, security logs, and private threat data provided by participating organizations. Upon collection, the data is analyzed, classified, and mapped against the **MITRE ATT&CK** framework to identify relevant tactics and techniques used by threat actors.

A **JSON file** containing the structured threat data is then generated, which includes indicators of compromise (IOCs), such as malicious IPs, file hashes, and URLs. This file is encrypted to ensure confidentiality during the sharing process.

3.2.2. Uploading and Sharing Data via IPFS

Once the threat intelligence is securely packaged into an encrypted JSON file, the first participant (User A) uploads the file to the **IPFS**. After the upload, **User A** receives a **Content Identifier (CID)**, a unique hash that acts as a pointer to the encrypted data on the IPFS.

User A then sends the CID to a trusted partner (User B) via a transaction submitted to **Hyperledger Fabric**. This step utilizes Hyperledger's chaincode and endorsement system to ensure that the transaction is validated and immutable. The CID is sent along with the sender's verified identity.

3.2.3. Authentication and Retrieval by Trusted Partner

Upon receiving the CID, **User B** must pass through an authentication check via the **chaincode** and **endorsement** processes in Hyperledger Fabric. This ensures that only authorized participants can access the shared threat intelligence data. Once authenticated, **User B** retrieves the CID and uses it to search for the encrypted file on the **IPFS**.

After retrieving the file, **User B** decrypts it to access the **JSON file**, which contains detailed threat information. The decrypted data is then visualized using the **MITRE ATT&CK Navigator**, allowing **User B** to understand the tactics, techniques, and procedures associated with the observed threats.

3.2.4. System Architecture and Workflow

The overall process of threat data sharing and retrieval is illustrated in the system architecture and workflow, as detailed in Figure 5. The architecture incorporates the use of **Hyperledger Fabric** for secure transactions and identity management, the **IPFS** for scalable decentralized file storage, and **MITRE ATT&CK** for structured threat analysis and visualization.

Each participant's node in the Hyperledger Fabric network is connected through secure channels, ensuring that only validated participants can interact with the data. The use of **chaincode** ensures that transactions are tamper-proof and enforce the necessary access controls.

3.2.5. Integration and Security Measures

The integration of **Hyperledger Fabric** and the **IPFS** facilitates the seamless exchange of threat intelligence data. The blockchain ensures data integrity and authentication through cryptographic proofs, while the IPFS allows for efficient off-chain storage. By utilizing encryption techniques such as **CP-ABE** (Ciphertext-Policy Attribute-Based Encryption), we ensure the confidentiality and integrity of the threat data.

This architecture introduces a novel decentralized approach to threat intelligence sharing, emphasizing security, scalability, and interoperability. The collaboration of Hyperledger Fabric and IPFS enables lightweight and efficient operations, reducing the computational load on the blockchain while maintaining robust security.

3.2.6. Scalability and Adaptability

The framework is designed with scalability in mind, accommodating a growing number of participants and increasing volumes of threat data. The modular nature of **Hyperledger Fabric** allows for easy integration with new partners and adaptation to different cybersecurity needs. Furthermore, the system's interoperability ensures compatibility with existing threat intelligence frameworks, such as **STIX** and **TAXII**, enhancing its applicability in real-world scenarios.

This methodology combines advanced cryptographic techniques, decentralized storage, and structured threat intelligence frameworks to create a robust and secure system for threat data sharing. By leveraging **Hyperledger Fabric**, the **IPFS**, and **MITRE ATT&CK**, our design ensures the efficient, scalable, and trustworthy exchange of threat intelligence data among organizations. This approach not only addresses current cybersecurity challenges but also sets the foundation for future innovations in the field of collaborative threat intelligence.

3.3. Hybrid Trust Model

TrustShare employs a hybrid trust model that combines *direct trust (DT)* and *indirect trust (IT)* to evaluate the reliability of participants dynamically.

Direct Trust (DT) is calculated from the history of interactions between participants, capturing metrics such as success rates of past information exchanges and responsiveness. This component reflects the firsthand experience that participant *i* has with participant *j*.

Indirect Trust (IT) aggregates feedback from other participants regarding participant *j*'s behavior. This feedback is weighted according to the trustworthiness of the feedback providers and incorporates a decay factor to ensure that recent behavior has greater influence than older assessments.

The overall trust score T_{ij} assigned by participant *i* to participant *j* is computed as a weighted sum of direct and indirect trust:

$$T_{ij} = \alpha \cdot DT_{ij} + (1 - \alpha) \cdot IT_{ij}$$

where

- T_{ij} is the combined trust score.

- $\alpha \in [0,1]$ is a tunable parameter balancing the influence of direct and indirect trust.
- DT_{ij} is the direct trust value based on past interactions.
- IT_{ij} is the indirect trust value derived from third-party feedback.

An example calculation is provided in Table 6, illustrating how DT_{ij} and IT_{ij} are combined using a chosen α.

Table 6. Example calculation of hybrid trust score.

Parameter	Value	Description
Direct Trust, DT_{ij}	0.75	Calculated from past interactions, success rates, and responsiveness.
Indirect Trust, IT_{ij}	0.60	Aggregated feedback from other participants, weighted by trustworthiness and time decay.
Weight Factor, α	0.7	Tunable parameter emphasizing direct trust over indirect trust.
Hybrid Trust Score, T_{ij}	0.705	Computed as $0.7 \times 0.75 + 0.3 \times 0.60$. Reflects combined trust level.

4. Evaluation and Performance Analysis

This section presents a comprehensive performance and security evaluation of the TrustShare infrastructure. The assessment focuses on architectural scalability, operational efficiency, security assurances, and comparative advantages over a centralized system. The infrastructure is implemented on Hyperledger Fabric and benchmarked using Hyperledger Caliper (v0.5.0), with performance indicators such as latency, throughput, and CPU utilization recorded under diverse deployment scenarios.

4.1. Scalability and Architectural Performance

TrustShare leverages Hyperledger Fabric's modular *execute–order–validate* model to decouple transaction processing from ordering and validation [31]. This architectural design enables parallelism, improving both throughput and latency by mitigating smart contract non-determinism and resource contention.

Latency performance of TrustShare is comparable to traditional systems like PostgreSQL, particularly under increased load [79]. Kubernetes-based orchestration enhances scalability through dynamic resource allocation, while IPFS integration ensures secure off-chain replication and data availability.

The transaction throughput is largely governed by the ordering service. Employing consensus mechanisms such as RAFT or Kafka allows horizontal scalability through node addition [33].

4.2. Benchmarking Methodology and Results

Benchmarking simulations included CTI ingestion (STIX format), policy revocation, and query operations across four deployment scenarios:

- Single organization, 2 peers.
- Two organizations, 4 peers.
- Kubernetes-based cluster deployment.
- Hyperledger Fabric with CouchDB as state database.

Figure 6 and Table 7 present the captured metrics. The Kubernetes configuration delivered the best performance, achieving 500 TPS with 75 ms latency and 72% CPU utilization.

Table 7. Performance metrics under various configurations.

Configuration	Latency (ms)	Throughput (TPS)	CPU Usage (%)
Single Organization, 2 Peers	90	180	40%
Two Organizations, 4 Peers	130	320	65%
Kubernetes Cluster	75	500	72%
With CouchDB	110	300	50%

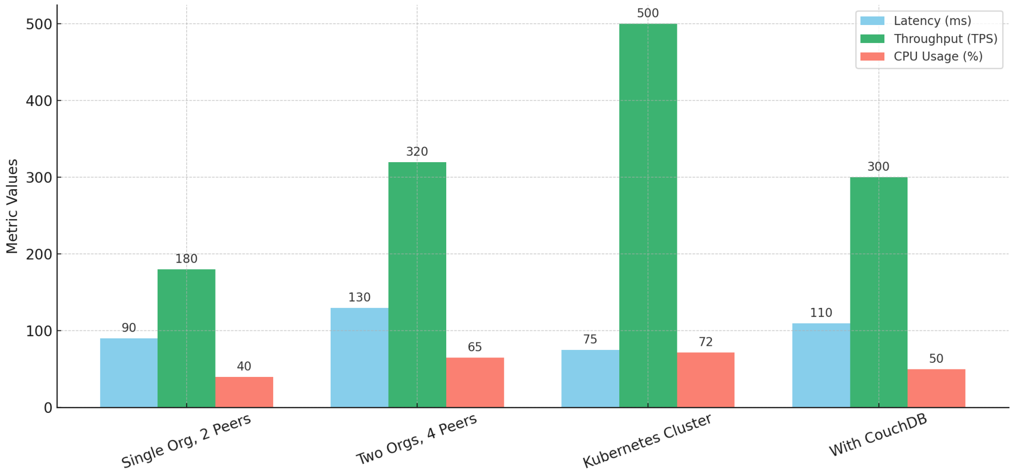

Figure 6. TrustShare performance metrics across deployment configurations.

4.3. Comparison with Centralized Baseline

To rigorously assess the advantages of a decentralized ledger-based architecture, TrustShare was evaluated in comparison with a conventional centralized system simulating a trusted intermediary. Identical transaction workloads were applied to both systems under controlled conditions. Metrics were averaged over 10 runs, capturing transaction latency, throughput, availability, and trust requirements. The corresponding results are illustrated in Figure 7 (latency under load), Figure 8 (throughput across nodes), Figure 9 (availability during node failures) and Figure 10 (comparative performance metrics), and summarized in Table 8.

Experimental Findings

- **Latency:** The centralized system has lower latency under light load, but performance degrades sharply under stress. TrustShare's latency increases more gradually, maintaining predictable performance, as presented in Figure 7.
- **Throughput and Scalability:** TrustShare scales with node count until consensus overhead gains plateau. The centralized model shows declining throughput as node count increases due to coordinator saturation. The results are presented in Figure 8.
- **Trust Model:** TrustShare distributes trust through consensus. The centralized system requires full trust in a single coordinator, introducing a critical vulnerability.
- **Availability and Fault Tolerance:** TrustShare maintains availability under node failures via Byzantine fault-tolerant mechanisms. In contrast, the centralized system fails after coordinator loss, as depicted in Figure 9.

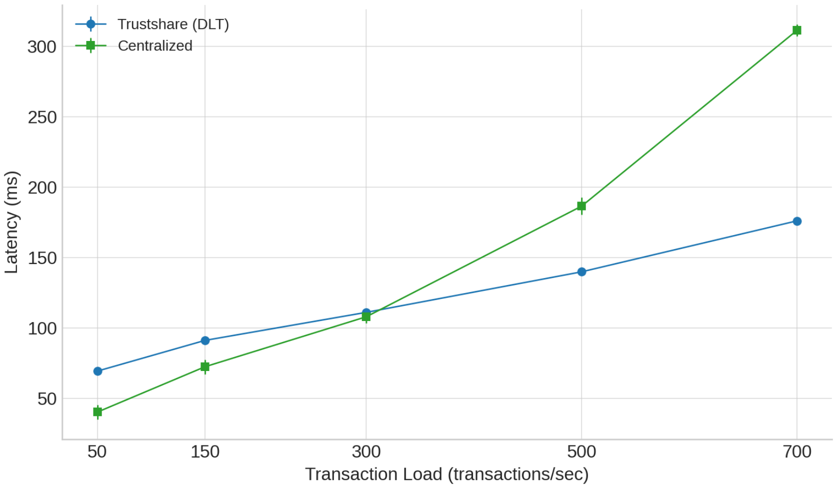

Figure 7. Latency under increasing load.

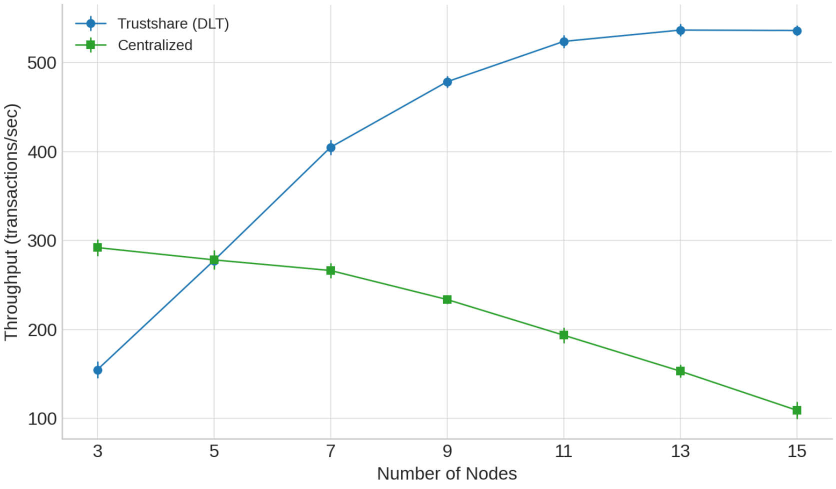

Figure 8. Throughput vs. node count.

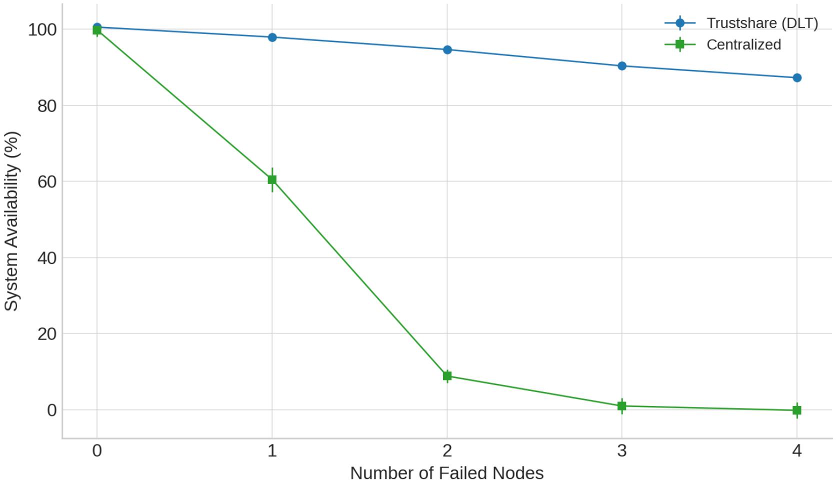

Figure 9. System availability vs. node failures.

Table 8. Comparison between TrustShare and centralized baseline.

Metric	TrustShare (DLT)	Centralized Baseline
Average Latency (ms)	70–175 (steady growth)	40–310 (non-linear spike)
Throughput (tx/sec)	150–540 (scalable)	300–110 (declines with load)
Fault Tolerance	Byzantine Fault Tolerance	Single Point of Failure
Trust Assumptions	Distributed validation	Full trust in coordinator
Scalability	High (until consensus limit)	Low (coordinator bottleneck)
Auditability	Immutable ledger	No native audit support
Deployment Complexity	Moderate–High	Low

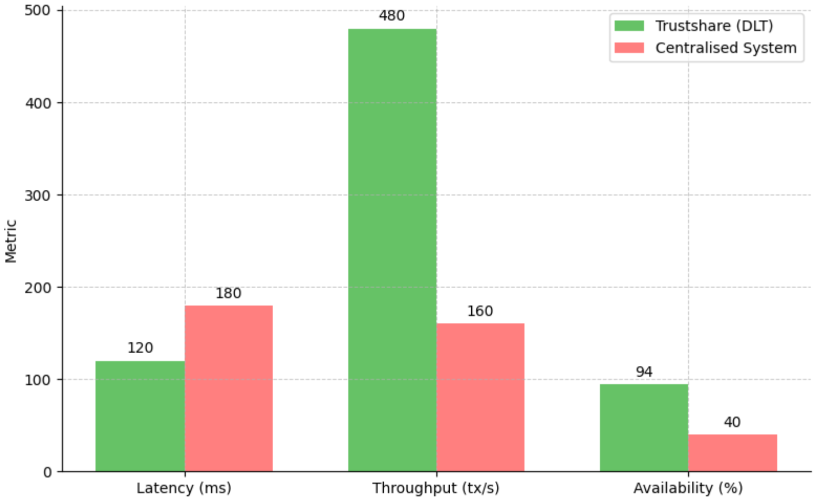

Figure 10. Summary of TrustShare vs. centralized baseline performance.

4.4. Security and Availability Analysis

Security is integral to TrustShare’s architecture, as summarized in Table 9. The distributed ledger design mitigates centralized attack vectors and denial-of-service threats. Hyperledger Fabric further enhances security by enforcing access control and cryptographic identities through its Membership Service Provider (MSP), ensuring end-to-end transaction confidentiality.

Table 9. Key security features in the TrustShare framework.

Security Layer	Mechanism Implemented
Identity Management	MSP via X.509 Certificates issued by Hyperledger CA
Data Encryption	Ciphertext-Policy Attribute-Based Encryption (CP-ABE)
Data Storage Security	IPFS-based encrypted off-chain storage
Access Control	Smart contract–enforced policy-based access
Anonymity	Optional ring signatures (FCsLRSs) for contributor privacy

TrustShare’s resilience is bolstered by node redundancy, fault-tolerant consensus, and revocable identity credentials via the certificate authority (CA). While the system defends against most infrastructure-level threats, further hardening is required against credential theft and social engineering.

4.5. Discussion and Roadmap

The evaluation confirms TrustShare’s effectiveness in delivering scalable, secure, and policy-aware CTI sharing. Future enhancements aim to improve usability, automation, and resilience:

- Integration with live CTI feeds and real-time analytics pipelines.

- Graphical policy management interface.
- Automated certificate renewal workflows.
- Enhanced auditing and threat simulation modules.
- Extended testing under adversarial network conditions.

These improvements will further strengthen TrustShare's readiness for operational deployment in collaborative cybersecurity environments.

5. Security Analysis

This section presents a structured evaluation of the security measures in the TrustShare framework, mapping each threat mitigation to operational steps in the threat intelligence (TI) lifecycle. Figure 5 illustrates the data flow, while Table 10 summarizes how specific threats are addressed at each stage.

Table 10. Threat mitigation mapping across TrustShare process steps.

Threat	Description	Mitigation Strategy	Enforced in Process Step(s)
Sybil Attack	Adversary creates fake identities to subvert the system.	Identity validation via MSP and CA-issued X.509 certificates; verified at user registration (pre-Step 1) and transaction submission.	Pre-Step 1; Step 7
Chaincode Poisoning	Injection of malicious or unauthorized chaincode.	Multi-peer endorsement policies for chaincode invocation; secure CI/CD pipeline with version control and audit logs.	Step 8
Identity Spoofing	Impersonation of authorized users or nodes.	Mutual TLS authentication and certificate validation enforced during access and data retrieval.	Steps 10–12
Replay Attack	Reuse of previously valid or fraudulent transactions.	Hyperledger Fabric-native timestamps and cryptographic nonces validated by ordering service and peers.	Steps 7–9
Insider Threats/Credential Misuse	Malicious actions by legitimate users or credential compromise.	CP-ABE encryption for fine-grained access control; immutable blockchain audit trails for accountability.	Steps 4, 9, 12, and 15
Anomalous Behavior	Suspicious or abnormal access patterns indicating compromise.	Planned SIEM integration for real-time monitoring; rule-based and ML anomaly detection on logs.	Steps 1–3, 12–15

5.1. Threat Model

TrustShare considers both external and internal adversaries:

- **External adversaries:** May attempt Sybil attacks, replay attacks, or impersonate authorized users or systems.
- **Internal adversaries:** May involve compromised insiders or misuse of credentials by legitimate users.

The system enforces the principles of confidentiality, integrity, availability, and auditability to counter these threats, in alignment with privacy regulations such as the GDPR.

5.2. Threat Mitigation

TrustShare uses a defense-in-depth approach, incorporating Ciphertext-Policy Attribute-Based Encryption (CP-ABE), Hyperledger Fabric's permissioned blockchain, and X.509 certificate-based identity management. These controls are implemented across the TrustShare workflow, as shown in Table 10.

Below is a brief explanation of how each mitigation aligns with specific threats:

1. **Sybil Attacks (Pre-Step 1; Step 7):** Only identities validated via Hyperledger Fabric's MSP and trusted certificate authorities are allowed to register and submit transactions.
2. **Chaincode Poisoning (Step 8):** Chaincode deployment requires multi-peer endorsements and is secured through a CI/CD pipeline with auditing and version control.
3. **Identity Spoofing (Steps 10–12):** TrustShare uses mutual TLS to ensure endpoint authenticity during data access and retrieval.
4. **Replay Attacks (Steps 7–9):** Hyperledger Fabric-native timestamps and cryptographic nonces ensure message freshness, mitigating replay attempts.
5. **Insider Threats and Credential Misuse (Steps 4, 9, 12, and 15):** CP-ABE enforces encryption-time access control, while blockchain audit logs provide tamper-evident traces.
6. **Anomalous Behavior (Steps 1–3, 12–15):** Planned integration with SIEM tools such as the ELK Stack and Splunk will support anomaly detection using rule-based and machine learning methods.

Future Work: While CP-ABE enforces robust encryption under the Decisional Bilinear Diffie–Hellman (DBDH) assumption, future work will investigate post-quantum cryptographic alternatives to safeguard long-term confidentiality. SIEM integration will also be operationalized in the production release.

This layered architecture, aligned with operational checkpoints and enforced at the protocol level, ensures TrustShare's resilience against common and advanced threat vectors while maintaining compliance and auditability.

5.3. Trust and Transparency

The endorsement policy in Hyperledger Fabric is defined by channel administrators and determines which peers must approve a transaction before it proceeds to the ordering service. Compliance with the policy ensures successful transaction submissions. Utilizing an endorsement policy that involves multiple peers from different organizations enhances transparency and trust, preventing single points of failure by setting a minimum number or percentage of endorsing peers.

The inclusion of "trust" reflects the understanding that effective threat intelligence sharing relies on establishing trust relationships between participating entities. Trust is a fundamental element in enabling secure and collaborative information exchange, implying the consideration of mechanisms to authenticate and validate the trustworthiness of the involved parties.

Hyperledger Fabric operates as a permissioned blockchain, wherein permissions are regulated by one or more membership service providers (MSPs) utilizing cryptographic identities. At each stage, transactions are meticulously examined to verify the authenticity of requests. This meticulous verification process serves to restrict unwanted access and fosters a heightened sense of trust in the system.

5.4. Privacy-Preserving Using Network Segmentation

Partitioning the network into distinct channels, each serving a specific purpose and involving a subset of organizations with their dedicated endorsers, offers a potential enhancement in performance by reducing the overall workload. This approach, akin to sharding in Hyperledger Fabric, allows for privacy and horizontal scaling, thereby addressing privacy and scalability concerns that have been widely discussed and proposed in the context of blockchain technology.

This method of preserving privacy emphasizes the need to protect sensitive information while sharing threat intelligence, acknowledging the significance of data confidentiality.

This suggests a focus on ensuring that privacy measures are in place to safeguard the shared information from unauthorized access.

The phrase "policy-based access control mechanisms" suggests the utilization of policies and rules to regulate access to shared threat intelligence. This indicates a proactive approach to control and manage the dissemination of information based on predefined policies, ensuring that access is granted only to authorized individuals or entities.

5.5. Network Authentication and Data Access Control

In our threat intelligence sharing framework, we employ a robust network authentication system by integrating Hyperledger Fabric with Ciphertext-Policy Attribute-Based Encryption (CP-ABE). Each participant is assigned unique attributes, and access to threat intelligence is governed by CP-ABE policies. Hyperledger Fabric's certificate authority manages participant certificates, ensuring secure identity authentication. Hyperledger Fabric's decentralized architecture enhances security by eliminating potential single points of failure. This integrated solution provides granular access control, reinforcing the confidentiality and trustworthiness of our threat intelligence sharing ecosystem.

5.6. Real-World Context for Attribute and Policy Complexity with Advanced Access Constraints

To underscore the importance of evaluating CP-ABE performance under increasing attribute cardinality and complex policy structures, as shown in Figure 11, we introduce two real-world cyber threat intelligence (CTI) sharing scenarios. These cases incorporate our proposed enforcement mechanisms: *time rivaling*, *time-bombing*, and *location interlocking*, which extend traditional CP-ABE capabilities with contextual awareness.

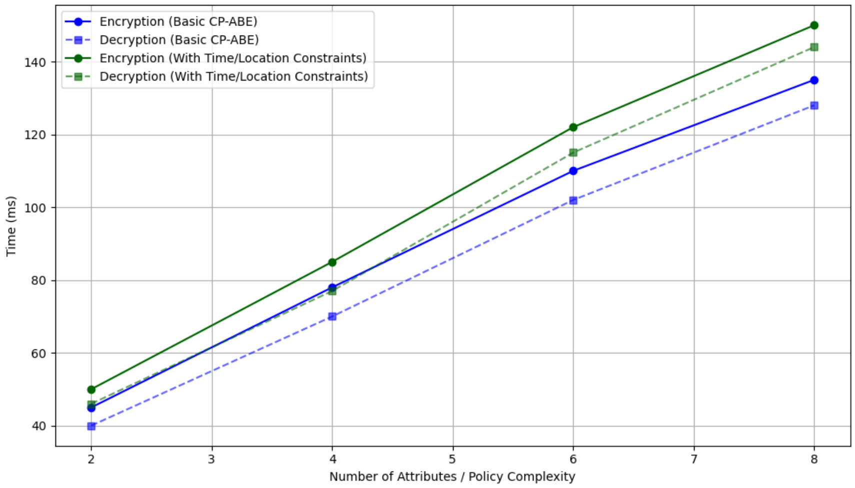

Figure 11. Performance of CP-ABE under complex policies incorporating *time rivaling*, *time-bombing*, and *location interlocking* across varying attribute counts and policy sizes.

Basic CP-ABE attributes are defined at the time of key issuance and do not change based on real-time conditions. This is what differentiates basic CP-ABE from context-aware or dynamic CP-ABE (which includes constraints like "Access only valid before 24 h" or "User must be in EU region").

Scenario 1: Healthcare Threat Sharing Consortium: In this use case, hospitals and certified diagnostic laboratories collaborate to exchange time-sensitive indicators of compromise (IoCs) related to ransomware campaigns. Access to shared data is governed by a multifaceted policy:

$$\texttt{Role: Analyst} \land \texttt{Org: Certified Lab} \land \texttt{Clearance: Level-2} \land \texttt{Access Time} < \texttt{48 h} \land \texttt{Location: US}$$

Here, **time-bombing** ensures that decryption keys expire 48 h after issuance, mitigating the risk of outdated intelligence usage. **Time rivaling** prioritizes and promotes newer threat reports, automatically suppressing older data. Meanwhile, **location interlocking** enforces decryption exclusively on authorized devices physically located in the United States, aligning with health data localization policies.

Scenario 2: Financial Sector Threat Intelligence Network: This network enables secure threat data exchange among banks and financial entities. For example, access to phishing and fraud-related intelligence is subject to the following access control policy:

$$\texttt{Entity Type: Bank} \land \texttt{Compliance Tier: Tier-1} \land \texttt{Subscription Level: Premium}$$
$$\land\, \texttt{Access Time < 24 h} \land \texttt{Location: EU}$$

In this scenario, **time-bombing** enforces ephemeral decryption keys with a 24 h lifespan. **Time rivaling** enables systems to give precedence to up-to-date threat data, and **location interlocking** ensures compliance with EU data residency requirements under the GDPR by restricting decryption to EU-based endpoints.

These scenarios highlight the applicability of expressive attribute-rich access policies in operational CTI environments. Our implementation demonstrates that CP-ABE can remain computationally feasible even when extended with contextual constraints, such as time and location. The complete encryption and decryption flow for this extended CP-ABE scheme is outlined in Algorithm 1.

Algorithm 1 Lattice-Based CP-ABE with Temporal and Geospatial Constraints

```
Require: Security parameter $\lambda$, Attribute universe $U$, Access policy $\mathcal{A}$, Message $M$, Timestamp $t$, Location $l$
Ensure: Ciphertext CT decryptable only by authorized users with valid temporal and location attributes
Setup($\lambda, U$)
Generate base matrix $A \in \mathbb{Z}_q^{n\times m}$ with associated trapdoor Trapdoor($A$)
Sample a short secret vector $\mathbf{s} \in \mathbb{Z}_q^n$
for each attribute attr $\in U$ do
    Generate a matrix $A_{\mathsf{attr}}$ and corresponding trapdoor
end for
Output public parameters PK and master secret key MSK
KeyGen(MSK, $S$)
for each attr $\in S$ do
    Generate short vector $\mathbf{d}_{\mathsf{attr}}$ using Trapdoor($A_{\mathsf{attr}}$)
end for
Add dynamic attributes: time < 24h, location: EU, freshness score
Output secret key SK $= \{\mathbf{d}_{\mathsf{attr}}\}$
Encrypt(PK, $M$, $\mathcal{A}$)
Represent access structure $\mathcal{A}$ as an LSSS matrix $(M, \rho)$, including temporal and spatial constraints
Sample random vector $\mathbf{v}$ and compute ciphertext components using LWE encryption
Output ciphertext CT = (components, $\mathcal{A}$)
Decrypt(SK, CT)
Verify that user's attributes satisfy $\mathcal{A}$, including checks on timestamp and geolocation
Use LSSS decoding vector to reconstruct the shared secret
Recover message $M$ only if policy evaluation passes
```

5.7. Anonymity Considerations

Due to the incorporation of the membership service provider (MSP) entity for identity management, Hyperledger Fabric inherently ensures transaction anonymity. The MSP is adaptable, allowing each organization to establish its own, contributing to pluggable data transaction anonymity. Transaction data anonymity is achieved through encryption using the client's private key, endorsed by a trusted third party via the Transport Layer Security (TLS) protocol, ensuring encrypted communication between participants.

Authorized users, acknowledged in the chaincode, must obtain adequate endorsements to validate transactions, adhering and subject to endorser rules. The study by [80] proposes an Anonymous Endorsement System with a threshold endorsement policy. This work introduces a novel ring signature scheme, Hyperledger Fabric's Constant-Sized Linkable Ring Signature (FCsLRS), ensuring transaction-oriented linkability for endorser identity concealment. The alteration of the RSA modulus size, implemented in Golang, undergoes rigorous examination for security and performance, with empirical studies supporting its viability. Notably, the rapid production of signatures and tags, coupled with a constant RSA modulus value, irrespective of message length or endorsement set size, underscores the collaborative creation of signatures by all endorsers.

6. Conclusions

This paper introduced *TrustShare*, a novel blockchain-based framework for secure, privacy-preserving, and regulation-compliant cyber threat intelligence (CTI) sharing. Based on a permissioned architecture powered by **Hyperledger Fabric**, TrustShare harnesses the strengths of distributed ledger technologies alongside **fine-grained cryptographic access control** to enable secure, scalable, and trustworthy collaboration across multiple organizations.

At the heart of TrustShare lies an advanced implementation of **Ciphertext-Policy Attribute-Based Encryption (CP-ABE)**, facilitating dynamic policy-driven data exchange. Moving beyond traditional attribute models, our framework incorporates **temporal ("time-bombing") constraints**, **controlled revelation mechanisms**, and **location interlocking**, allowing data owners to stipulate precisely when, how, and where shared intelligence may be accessed. These enhancements provide unprecedented control over sensitive threat data whilst maintaining operational flexibility and adherence to regulatory frameworks.

To ensure interoperability and applicability in real-world settings, the framework integrates the **STIX and TAXII** standards for structured threat data representation and employs the **IPFS** for secure decentralized storage. In compliance with the **General Data Protection Regulation (GDPR)** and the **right to be forgotten (RtbF)**, the design embeds privacy-preserving data policies and revocation capabilities, enforced through chaincode and certificate authority mechanisms.

Empirical assessments indicate that TrustShare achieves low-latency performance, scalable throughput, and high resource efficiency across various deployment environments, including containerized infrastructures orchestrated with Kubernetes. Furthermore, the security architecture—comprising role-based endorsements, endorsement anonymity (via FCsLRS), and network segmentation—demonstrates strong resilience against common attack vectors and insider threats.

Nevertheless, several challenges persist. The computational cost associated with complex CP-ABE policies and encrypted operations may impact scalability under real-time demands. The current framework assumes a partially trusted model among organizational participants. Therefore, establishing comprehensive inter-domain trust governance remains a significant avenue for further exploration.

Future work will focus on enhancing TrustShare's adaptability, efficiency, and long-term resilience. This includes the incorporation of context-aware access controls driven by dynamic threat scoring to enable real-time policy adjustments, and improvements to policy enforcement and compilation mechanisms to support scalability in attribute-rich environments. Additionally, the integration of privacy-preserving artificial intelligence is envisioned to facilitate collaborative threat prioritization and anomaly detection across distributed actors. To ensure future-proof confidentiality, ongoing research will also explore post-quantum cryptographic alternatives to the current CP-ABE scheme, protecting against emerging quantum-capable adversaries.

Field trials within national and sector-specific CTI alliances are planned to further evaluate the robustness, practicality, and compliance of the system under operational conditions.

In summary, TrustShare exemplifies how the synthesis of permissioned blockchain, advanced cryptographic policy control, and decentralized storage can support the creation of a **trustworthy, sovereign, and regulation-aligned** platform for the next generation of cyber threat intelligence sharing.

Author Contributions: H.A., Project administration, Conceptualization, Methodology, Formal analysis, Investigation, Software development, Writing—original draft, Writing—review and editing, Validation, Resources. W.J.B., Supervision, Validation, Review and editing, Resources. J.A., Supervision, Validation, Review. M.A., Validation, Review. M.S.K., Validation, Review. I.W., Validation, Review. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article material. Further inquiries can be directed to the corresponding authors.

Conflicts of Interest: The authors declare that they don't have any conflicting interests.

References

1. Kumar, P.; Wazid, M.; Singh, D.; Singh, J.; Das, A.K.; Park, Y.; Rodrigues, J.J. Explainable artificial intelligence envisioned security mechanism for cyber threat hunting. *Secur. Priv.* **2023**, *6*, e312. [CrossRef]
2. Ma, J.; Jiao, W.; Gao, H. A Study on ChinaBond Threat Intelligence Platform Construction. *Bond Mon.* **2022**. [CrossRef]
3. CrowdStrike. Threat Intelligence: What It Is, Types and Benefits. Available online: https://www.crowdstrike.com/en-us/cybersecurity-101/threat-intelligence/#:~:text=Threat%20intelligence%20refers%20to%20the,informed%2C%20data%2Ddriven%20decisions. (accessed on 30 March 2025).
4. SANS Institute. What Is Threat Intelligence? 2016. Available online: https://www.sans.org/white-papers/ (accessed on 30 December 2024).
5. Trend Micro. *Navigating New Frontiers: Trend Micro 2021 Annual Cybersecurity Report*; Technical report, analyses major cybersecurity trends from 2021; Trend Micro: Tokyo, Japan, 2022.
6. Wagner, T.D.; Mahbub, K.; Palomar, E.; Abdallah, A.E. Cyber threat intelligence sharing: Survey and research directions. *Comput. Secur.* **2019**, *87*, 101589. [CrossRef]
7. Gao, S.; Piao, G.; Zhu, J.; Ma, X.; Ma, J. TrustAccess: A Trustworthy Secure Ciphertext-Policy and Attribute Hiding Access Control Scheme Based on Blockchain. *IEEE Trans. Veh. Technol.* **2020**, *69*, 5784–5798. [CrossRef]
8. Ramsdale, A.; Shiaeles, S.; Kolokotronis, N. A comparative analysis of cyber-threat intelligence sources, formats and languages. *Electronics* **2020**, *9*, 824. [CrossRef]
9. Sauerwein, C.; Fischer, D.; Rubsamen, M.; Rosenberger, G.; Stelzer, D.; Breu, R. From threat data to actionable intelligence: An exploratory analysis of the intelligence cycle implementation in cyber threat intelligence sharing platforms. In Proceedings of the 16th International Conference on Availability, Reliability and Security, Vienna, Austria, 17–20 August 2021; pp. 1–9.
10. Guarascio, M.; Cassavia, N.; Pisani, F.S.; Manco, G. Boosting cyber-threat intelligence via collaborative intrusion detection. *Future Gener. Comput. Syst.* **2022**, *135*, 30–43. [CrossRef]
11. Johnson, C.; Badger, L.; Waltermire, D.; Snyder, J.; Skorupka, C. Guide to Cyber Threat Information Sharing. NIST Special Publication800-150. 2016. Available online: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-150.pdf (accessed on 26 April 2025).
12. Cho, H. ASIC-Resistance of Multi-Hash Proof-of-Work Mechanisms for Blockchain Consensus Protocols. *IEEE Access* **2018**, *6*, 66210–66222. [CrossRef]
13. Singh, S.; Singh, N. Blockchain: Future of Financial and Cyber Security. In Proceedings of the 2nd International Conference on Contemporary Computing and Informatics (IC3I), Greater Noida, India, 14–17 December 2016; pp. 463–467. [CrossRef]
14. Nakamoto, S. *Bitcoin: A Peer-to-Peer Electronic Cash System*; Bitcoin.org: Online, 2008. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 26 April 2025).

15. Judmayer, A.; Stifter, N.; Krombholz, K.; Weippl, E. *Blocks and Chains: Introduction to Bitcoin, Cryptocurrencies, and Their Consensus Mechanisms*; Springer Nature: Berlin/Heidelberg, Germany, 2022.
16. Lepore, C.; Ceria, M.; Visconti, A.; Rao, U.P.; Shah, K.A.; Zanolini, L. A survey on blockchain consensus with a performance comparison of PoW, PoS and pure PoS. *Mathematics* **2020**, *8*, 1782. [CrossRef]
17. Dunnett, K.; Pal, S.; Jadidi, Z. Challenges and Opportunities of Blockchain for Cyber Threat Intelligence Sharing. In *Secure and Trusted Cyber Physical Systems: Recent Approaches and Future Directions*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 1–24.
18. Abubakar, M.; Ali, H.; Ghaleb, B.; Wadhaj, I.; Buchanan, W.J. An Overview of Blockchain-Based IoT Architectures and Designs. In Proceedings of the International Conference on Emerging Technologies and Intelligent Systems, Cham, Switzerland, 2–3 September 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 596–605.
19. Wüst, K.; Gervais, A. Do you Need a Blockchain? In Proceedings of the Crypto Valley Conference on Blockchain Technology (CVCBT), Zug, Switzerland, 20–22 June 2018; pp. 45–54. [CrossRef]
20. Androulaki, E.; Barger, A.; Bortnikov, V.; Cachin, C.; Christidis, K.; De Caro, A.; Enyeart, D.; Ferris, C.; Laventman, G.; Manevich, Y.; et al. Hyperledger fabric: A distributed operating system for permissioned blockchains. In Proceedings of the Thirteenth EuroSys Conference, Porto, Portugal, 23–26 April 2018; pp. 1–15.
21. Hearn, M.; Brown, R.G. Corda: A Distributed Ledger. Corda Technical White Paper. 2016. Available online: https://docs.r3.com/en/pdf/corda-technical-whitepaper.pdf (accessed on 13 April 2025).
22. Özdemir, A. Cyber Threat Intelligence Sharing Technologies and Threat Sharing Model Using Blockchain. Master's Thesis, Middle East Technical University, Ankara, Türkiye, 2021.
23. Wood, G. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Proj. Yellow Pap.* **2014**, *151*, 1–32.
24. Valenta, M.; Sandner, P. Comparison of ethereum, hyperledger fabric and corda. *Frankf. Sch. Blockchain Cent.* **2017**, *8*, 1–8.
25. Polge, J.; Robert, J.; Le Traon, Y. Permissioned blockchain frameworks in the industry: A comparison. *Ict Express* **2021**, *7*, 229–233. [CrossRef]
26. Dinh, T.T.A.; Wang, J.; Chen, G.; Liu, R.; Ooi, B.C.; Tan, K.L. Blockbench: A framework for analyzing private blockchains. In Proceedings of the ACM International Conference on Management of Data, Chicago, IL, USA, 14–19 May 2017; pp. 1085–1100.
27. Brandenburger, M.; Cachin, C.; Kapitza, R.; Sorniotti, A. Blockchain and trusted computing: Problems, pitfalls, and a solution for hyperledger fabric. *arXiv* **2018**, arXiv:1805.08541.
28. Gorenflo, C.; Lee, S.; Golab, L.; Keshav, S. FastFabric: Scaling hyperledger fabric to 20 000 transactions per second. *Int. J. Netw. Manag.* **2020**, *30*, e2099. [CrossRef]
29. Nasir, Q.; Qasse, I.A.; Abu Talib, M.; Nassif, A.B. Performance Analysis of Hyperledger Fabric Platforms. *Secur. Commun. Netw.* **2018**, *2018*, 7431475. [CrossRef]
30. Punathumkandi, S.; Meenakshi, V. A deep dive into Hyperledger. In *Blockchain and Machine Learning for e-Healthcare Systems*; Institution of Engineering and Technology: London, UK, 2020; p. 85.
31. Ali, H.; Ahmad, J.; Jaroucheh, Z.; Papadopoulos, P.; Pitropakis, N.; Lo, O.; Abramson, W.; Buchanan, W.J. Trusted Threat Intelligence Sharing in Practice and Performance Benchmarking through the Hyperledger Fabric Platform. *Entropy* **2022**, *24*, 1379. [CrossRef]
32. Thakkar, P.; Nathan, S.; Viswanathan, B. Performance Benchmarking and Optimizing Hyperledger Fabric Blockchain Platform. In Proceedings of the IEEE 26th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Milwaukee, WI, USA, 25–28 September 2018; pp. 264–276. [CrossRef]
33. Fabric, H. Hyperledger Fabric Foundation. Available online: https://hyperledger-fabric.readthedocs.io/en/latest/test_network.html (accessed on 30 March 2025).
34. Bogatov, D.; De Caro, A.; Elkhiyaoui, K.; Tackmann, B. Anonymous transactions with revocation and auditing in hyperledger fabric. In Proceedings of the Cryptology and Network Security: 20th International Conference, CANS 2021, Vienna, Austria, 13–15 December 2021; pp. 435–459.
35. Homan, D.; Shiel, I.; Thorpe, C. A New Network Model for Cyber Threat Intelligence Sharing using Blockchain Technology. In Proceedings of the 10th IFIP International Conference on New Technologies, Mobility and Security (NTMS), Canary Islands, Spain, 24–26 June 2019; pp. 1–6. [CrossRef]
36. Zhang, X.; Miao, Y. A reputation-based blockchain model for secure and efficient cyber threat intelligence sharing. *J. Netw. Comput. Appl.* **2021**, *178*, 102985.
37. Nguyen, K.; Pal, S.; Jadidi, Z.; Dorri, A.; Jurdak, R. A Blockchain-Enabled Incentivised Framework for Cyber Threat Intelligence Sharing in ICS. In Proceedings of the IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops), Pisa, Italy, 21–25 March 2022; pp. 261–266. [CrossRef]
38. Ma, J.; Wang, Z.; Qiu, Y.; Liu, T.; Zhang, Z.; Li, H.; Chen, J.; Sun, W.; Zhao, J.; Xu, L.; et al. Blockchain-based incentive mechanism for cybersecurity information sharing: An evolutionary game approach. *Future Gener. Comput. Syst.* **2023**, *140*, 239–251.
39. Hu, J.; Zhu, P.; Li, J.; Qi, Y.; Xia, Y.; Wang, F.Y. A Secure Medical Information Storage and Sharing Method Based on Multiblockchain Architecture. *IEEE Trans. Comput. Soc. Syst.* **2024**, *11*, 6392–6406. [CrossRef]

40. Ranjan, S.; Negi, A.; Jain, H.; Pal, B.; Agrawal, H. Network System Design using Hyperledger Fabric: Permissioned Blockchain Framework. In Proceedings of the Twelfth International Conference on Contemporary Computing (IC3), Noida, India, 8–10 August 2019; pp. 1–6. [CrossRef]
41. Iftekhar, A.; Cui, X.; Tao, Q.; Zheng, C. Hyperledger fabric access control system for internet of things layer in blockchain-based applications. *Entropy* **2021**, *23*, 1054. [CrossRef]
42. Gourisetti, S.N.G.; Sebastian-Cardenas, D.J.; Bhattarai, B.; Wang, P.; Widergren, S.; Borkum, M.; Randall, A. Blockchain smart contract reference framework and program logic architecture for transactive energy systems. *Appl. Energy* **2021**, *304*, 117860. [CrossRef]
43. Ali, H.; Abubakar, M.; Ahmad, J.; Buchanan, W.J.; Jaroucheh, Z. PASSION: Permissioned Access Control for Segmented Devices and Identity for IoT Networks. In Proceedings of the IEEE 28th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), Edinburgh, UK, 6–8 November 2023; pp. 200–205. [CrossRef]
44. Brotsis, S.; Kolokotronis, N.; Limniotis, K.; Bendiab, G.; Shiaeles, S. On the Security and Privacy of Hyperledger Fabric: Challenges and Open Issues. In Proceedings of the IEEE World Congress on Services (SERVICES), Beijing, China, 18–23 October 2020; pp. 197–204. [CrossRef]
45. Ali, H.; Papadopoulos, P.; Ahmad, J.; Pitropakis, N.; Jaroucheh, Z.; Buchanan, W.J. Privacy-preserving and Trusted Threat Intelligence Sharing using Distributed Ledgers. In Proceedings of the 14th International Conference on Security of Information and Networks (SIN), Edinburgh, UK, 15–17 December 2021; Volume 1, pp. 1–6. [CrossRef]
46. Xu, X.; Sun, G.; Luo, L.; Cao, H.; Yu, H.; Vasilakos, A.V. Latency performance modeling and analysis for hyperledger fabric blockchain network. *Inf. Process. Manag.* **2021**, *58*, 102436. [CrossRef]
47. Caliper, H. Caliper—Blockchain Performance Benchmark Tool. 2025. Available online: https://github.com/hyperledger-caliper/caliper (accessed on 30 March 2025).
48. Haque, M.F.; Krishnan, R. Toward automated cyber defense with secure sharing of structured cyber threat intelligence. *Inf. Syst. Front.* **2021**, *23*, 883–896. [CrossRef]
49. Kumar, R.; Goyal, R. On cloud security requirements, threats, vulnerabilities and countermeasures: A survey. *Comput. Sci. Rev.* **2019**, *33*, 1–48. [CrossRef]
50. Deepa, N.; Pham, Q.V.; Nguyen, D.C.; Bhattacharya, S.; Prabadevi, B.; Gadekallu, T.R.; Maddikunta, P.K.R.; Fang, F.; Pathirana, P.N. A survey on blockchain for big data: Approaches, opportunities, and future directions. *Future Gener. Comput. Syst.* **2022**, *131*, 209–226. [CrossRef]
51. Riesco, R.; Villagrá, V.A. Leveraging cyber threat intelligence for a dynamic risk framework: Automation by using a semantic reasoner and a new combination of standards (STIX™, SWRL and OWL). *Int. J. Inf. Secur.* **2019**, *18*, 715–739. [CrossRef]
52. Riesco, R.; Larriva-Novo, X.; Villagrá, V.A. Cybersecurity threat intelligence knowledge exchange based on blockchain: Proposal of a new incentive model based on blockchain and Smart contracts to foster the cyber threat and risk intelligence exchange of information. *Telecommun. Syst.* **2020**, *73*, 259–288. [CrossRef]
53. Nour, B.; Pourzandi, M.; Debbabi, M. A Survey on Threat Hunting in Enterprise Networks. *IEEE Commun. Surv. Tutor.* **2023**, *25*, 2299–2324. [CrossRef]
54. Pahlevan, M.; Voulkidis, A.; Velivassaki, T.H. Secure exchange of cyber threat intelligence using TAXII and distributed ledger technologies-application for electrical power and energy system. In Proceedings of the 16th International Conference on Availability, Reliability and Security, Vienna, Austria, 17–20 August 2021; pp. 1–8.
55. Zutshi, A.; Grilo, A.; Nodehi, T. The value proposition of blockchain technologies and its impact on Digital Platforms. *Comput. Ind. Eng.* **2021**, *155*, 107187. [CrossRef]
56. Xue, L.; Yu, Y.; Li, Y.; Au, M.H.; Du, X.; Yang, B. Efficient attribute-based encryption with attribute revocation for assured data deletion. *Inf. Sci.* **2019**, *479*, 640–650. [CrossRef]
57. Morales-Sandoval, M.; Cabello, M.H.; Marin-Castro, H.M.; Compean, J.L.G. Attribute-Based Encryption Approach for Storage, Sharing and Retrieval of Encrypted Data in the Cloud. *IEEE Access* **2020**, *8*, 170101–170116. [CrossRef]
58. Zhang, G.; Chen, X.; Feng, B.; Guo, X.; Hao, X.; Ren, H.; Dong, C.; Zhang, Y. BCST-APTS: Blockchain and CP-ABE empowered data supervision, sharing, and privacy protection scheme for secure and trusted agricultural product traceability system. *Secur. Commun. Netw.* **2022**, *2022*, 1–11. [CrossRef]
59. Fugkeaw, S. A Fine-Grained and Lightweight Data Access Control Model for Mobile Cloud Computing. *IEEE Access* **2021**, *9*, 836–848. [CrossRef]
60. Porwal, S.; Mittal, S. A fully flexible key delegation mechanism with efficient fine-grained access control in CP-ABE. *J. Ambient Intell. Humaniz. Comput.* **2023**, *14*, 12837–12856. [CrossRef]
61. Das, S.; Namasudra, S. Multiauthority CP-ABE-based Access Control Model for IoT-enabled Healthcare Infrastructure. *IEEE Trans. Ind. Inform.* **2023**, *19*, 821–829. [CrossRef]
62. Zhang, L.; Cui, Y.; Mu, Y. Improving Security and Privacy Attribute Based Data Sharing in Cloud Computing. *IEEE Syst. J.* **2020**, *14*, 387–397. [CrossRef]

63. Yin, Y.; Gan, Q.; Zuo, C.; Liu, N.; Wang, C.; Jiang, Y. A Revocable Outsourced Data Accessing Control Scheme with Black-Box Traceability. In *Information Security Practice and Experience, Proceedings of the 19th International Conference, Wuhan, China, 25–27 October 2024*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 380–398.
64. Makhdoom, I.; Zhou, I.; Abolhasan, M.; Lipman, J.; Ni, W. PrivySharing: A blockchain-based framework for privacy-preserving and secure data sharing in smart cities. *Comput. Secur.* **2020**, *88*, 101653. [CrossRef]
65. Bhuvana, R.; Aithal, P. Blockchain based service: A case study on IBM blockchain services & hyperledger fabric. *Int. J. Case Stud. Bus. IT Educ.* **2020**, *4*, 94–102.
66. Khan, L.U.; Yaqoob, I.; Tran, N.H.; Kazmi, S.M.A.; Dang, T.N.; Hong, C.S. Edge-Computing-Enabled Smart Cities: A Comprehensive Survey. *IEEE Internet Things J.* **2020**, *7*, 10200–10232. [CrossRef]
67. Deng, H.; Meng, X.; Guo, J.; Xi, E.; Zhao, H. A Framework of Blockchain-Based Security for WBANs. In Proceedings of the 3rd International Conference on Smart BlockChain (SmartBlock), Zhengzhou, China, 23–25 October 2020; pp. 75–80. [CrossRef]
68. Banerjee, S.; Bera, B.; Das, A.K.; Chattopadhyay, S.; Khan, M.K.; Rodrigues, J.J. Private blockchain-envisioned multi-authority CP-ABE-based user access control scheme in IIoT. *Comput. Commun.* **2021**, *169*, 99–113. [CrossRef]
69. Casino, F.; Politou, E.; Alepis, E.; Patsakis, C. Immutability and Decentralized Storage: An Analysis of Emerging Threats. *IEEE Access* **2020**, *8*, 4737–4744. [CrossRef]
70. Krushnarao, W.A.; Gandage, S. Secure-Medishare: A Comprehensive Secure Medical Data-Sharing System Using Blockchain, Watermarking, Steganography, And Optimized Hybrid Cryptography. *Scand. J. Inf. Syst.* **2023**, *35*, 1–13.
71. Naz, M.; Al-zahrani, F.A.; Khalid, R.; Javaid, N.; Qamar, A.M.; Afzal, M.K.; Shafiq, M. A secure data sharing platform using blockchain and interplanetary file system. *Sustainability* **2019**, *11*, 7054. [CrossRef]
72. Kang, P.; Yang, W.; Zheng, J. Blockchain private file storage-sharing method based on IPFS. *Sensors* **2022**, *22*, 5100. [CrossRef] [PubMed]
73. Argyropoulos, V.; Alepis, E.; Patsakis, C. Semi-Decentralized File Sharing as a Service. In Proceedings of the 13th International Conference on Information, Intelligence, Systems & Applications (IISA), Corfu, Greece, 18–20 July 2022; pp. 1–8. [CrossRef]
74. Bhattacharya, P.; Saraswat, D.; Savaliya, D.; Sanghavi, S.; Verma, A.; Sakariya, V.; Tanwar, S.; Sharma, R.; Raboaca, M.S.; Manea, D.L. Towards future internet: The metaverse perspective for diverse industrial applications. *Mathematics* **2023**, *11*, 941. [CrossRef]
75. Trautwein, D.; Raman, A.; Tyson, G.; Castro, I.; Scott, W.; Schubotz, M.; Gipp, B.; Psaras, Y. Design and evaluation of IPFS: A storage layer for the decentralized web. In Proceedings of the ACM SIGCOMM 2022 Conference, Amsterdam, The Netherlands, 22–26 August 2022; pp. 739–752.
76. Kumar, S.; Bharti, A.K.; Amin, R. Decentralized secure storage of medical records using Blockchain and IPFS: A comparative analysis with future directions. *Secur. Priv.* **2021**, *4*, e162. [CrossRef]
77. Grundstrom, C.; Väyrynen, K.; Iivari, N.; Isomursu, M. Making sense of the general data protection regulation—Four categories of personal data access challenges. In Proceedings of the 52nd Hawaii International Conference on System Sciences, Maui, HI, USA, 8–11 January 2019.
78. Havelange, A.; Dumontier, M.; Wouters, B.; Linde, J.; Townend, D.; Riedl, A.; Urovi, V. LUCE: A Blockchain Solution for monitoring data License accoUntability and CompliancE. *arXiv* **2019**, arXiv:1908.02287.
79. Stamatellis, C.; Papadopoulos, P.; Pitropakis, N.; Katsikas, S.; Buchanan, W.J. A privacy-preserving healthcare framework using hyperledger fabric. *Sensors* **2020**, *20*, 6587. [CrossRef]
80. Mazumdar, S.; Ruj, S. Design of Anonymous Endorsement System in Hyperledger Fabric. *IEEE Trans. Emerg. Top. Comput.* **2021**, *9*, 1780–1791. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.