International Journal of Information Security (2025) 24:130
https://doi.org/10.1007/s10207-025-01052-w

REGULAR CONTRIBUTION

Anonymous provision of privacy-sensitive services using blockchain and decentralised storage

Stanisław Barański[1] · Julian Szymański[1] · Higinio Mora[2]

Abstract

Lawyers, laboratories, auditors, and banks often need access to sensitive personal data to provide services such as genetic testing, paternity testing, STD testing, credit scoring, or legal advice. Processing such data exposes both service providers (SPs) and users to privacy risks: SPs risk violating laws like the General Data Protection Regulation (GDPR) and the Consumer Protection Act (CPA), while users risk losing their privacy. We observe that personal data is often only needed for logistical purposes like payment or communication and could be provided anonymously if suitable methods existed. To address this, we present a solution that enables services to be delivered without collecting personal data. Our protocol combines anonymous payment methods (e.g., cash, privacy-preserving cryptocurrencies), blockchain for fairness, and distributed content-addressable storage networks to deliver results. Compared to existing approaches, our protocol achieves anonymity under weaker assumptions, supports the transfer of physical materials and conflict resolution, and eliminates the need for customer interaction with a trusted arbiter in conflict-free cases-making it more practical. We analyze the protocol's fairness and implement a prototype using Ethereum as a message board, Monero for anonymous payments, and Powergate (IPFS/Filecoin) as a decentralized storage solution.

Keywords Anonymity · Blockchain · Diagnosis · e-commerce · Fair-exchange · Privacy · Services

1 Introduction

Providing services in sensitive domains like healthcare, legal advice, or financial auditing increasingly relies on secure information handling. These services often require customers to share personal data, including sensitive information such as health records, biological materials, or confidential legal documents. This necessity exposes both users and service providers (SPs) to significant information security and privacy risks. SPs face the challenge of adhering to regulations like GDPR, CPA, and other data protection laws, while customers are vulnerable to privacy breaches, identity theft, and misuse of their sensitive data [1].

For individuals in the public eye, such as influencers, politicians, and celebrities, these risks are amplified. Exposure of their health records, financial transactions, or legal documents can severely damage their reputation or become tools for blackmail. The problem is further compounded when personal information is directly linked to sensitive records or biological materials. This can deter customers from seeking necessary services due to privacy concerns, negatively impacting both the customer and the service provider (SP) [2, 3].

Within healthcare systems, Electronic Medical Records (EMRs) are prime examples of sensitive data requiring stringent security measures [4, 5]. Privacy-preserving data mining (PPDM) is a vital research area, focusing on anonymization and secure patient-healthcare interactions [6–8]. Studies suggest that EMR adoption can inadvertently compromise confidentiality [3], and patient distrust due to privacy concerns can lead to the exclusion of genetic data from EMRs [2].

✉ Stanisław Barański
stanislaw.baranski@pg.edu.pl

Julian Szymański
julian.szymanski@pg.edu.pl

Higinio Mora
hmora@ua.es

[1] Department of Electronic, Telecommunication and Informatics, Gdansk University of Technology, Narutowicza 11/12, 80-233 Gdansk, Poland

[2] Department of Computer Science Technology and Computation, University of Alicante, San Vicente del Raspeig, 03690 Alicante, Spain

Published online: 17 May 2025

Our study explores the provision of services while minimizing the collection of personal information, thereby addressing privacy concerns proactively, even before data is entered into systems like EMRs. By maintaining customer anonymity during service delivery, we aim to reduce the level of trust required from customers and alleviate the responsibility borne by SPs regarding personal data protection. Key participants in this model include customers, SPs, and dispute resolution services. Examples of services that particularly benefit from anonymity include:

- **Confidentiality in Medical Testing**: Patients undergoing sensitive tests such as for drugs, STDs, paternity, or steroids benefit from privacy to avoid potential stigma or unwanted disclosure.
- **Anonymity in Legal Consultations for Entrepreneurs**: Entrepreneurs seeking risk assessment may prefer anonymous consultations to protect sensitive business ideas and prevent potential misuse of disclosed information.
- **Whistleblowing Platforms**: Anonymous reporting mechanisms for unethical or illegal activities foster transparency and accountability without fear of retaliation.
- **Sexual Health and Reproductive Services**: Individuals seeking advice on sensitive health topics prefer anonymity to avoid societal judgment or discrimination.

Using Solove's Taxonomy of Privacy [9], our protocol aims to prevent privacy risks such as breach of confidentiality, disclosure, identification, and secondary use. We consider adversaries like malicious insiders, government agencies, third-party services, hackers, and cybercriminals. Our proposed protocol aims to decouple personal information from materials, payment, and communication processes, thereby enhancing anonymity even under adversarial conditions.

However, designing a protocol that effectively hides the customer's identity while ensuring secure and coordinated service delivery presents significant challenges. In particular, resolving conflicts between customers and service providers becomes complex when customer identities are not known.

We address the challenge of anonymous service provision by framing it as a fair exchange problem. This approach ensures that in a transaction, either both parties (customer and service provider) receive what they expect, or neither does. In our specific context, this involves the exchange of money and sensitive (possibly physical) materials for a service result. Our review of existing systems indicates that none fully satisfy all our requirements, especially regarding anonymity and handling of physical materials with dispute resolution. We, therefore, propose a novel anonymous protocol tailored for services that require physical materials, incorporating robust dispute resolution mechanisms. In situations of conflict, our protocol allows customers to disclose relevant interaction details and provide verifiable proofs of SP misbehavior to dispute resolution services.

Our protocol is specifically designed for the commerce sector, particularly for service providers who handle sensitive and potentially physical materials provided by customers, such as biological samples, legal documents, or financial records. This includes applications in anonymous genetic testing, forensic analysis, and confidential research studies where personal data and physical materials must be handled with utmost privacy. Additionally, the protocol is beneficial for discreet services such as substance testing, private counseling, and therapy, as well as in sectors like industrial testing, environmental analysis, and intellectual property verification, where anonymity and secure handling of physical materials are crucial. These services span across industries such as healthcare, legal services, environmental science, and industrial compliance, highlighting the protocol's versatility in ensuring both privacy and compliance with legal regulations. By addressing the unique challenges of transactions involving physical materials, our approach ensures that these services can be delivered securely and fairly, even when the customer's identity remains protected.

To achieve this, our protocol integrates:

- **Blockchain**, to achieve fairness, i.e., as a means of proving that certain actions took place at a certain time without the Trusted Third Party (TTP).
- **Anonymous payment methods**, like cash or cryptocurrencies, for anonymous transactions.
- **Decentralised storage network** (e.g., IPFS) to enable secure delivery of results, coupled with a provable storage network (e.g., Filecoin) to guarantee result availability to the customer.
- **Cryptographic techniques**, including symmetric encryption, Diffie-Hellman key exchange, and digital signatures for secure communication and authentication.

Figure 1 presents a simplified protocol diagram. The contributions of this paper are:

- We propose a protocol based on a realistic operational and threat model that:
  - Enables **anonymous service provision** involving physical materials, thus eliminating the need for service providers to collect personal information.
  - Achieves **weak fairness** through blockchain technology and cryptographic proofs, modeled as an interactive non-cooperative game, ensuring fairness at each step, with recourse to dispute resolution.
  - In dispute-free cases operates without a centralized Trusted Third Party (TTP), leveraging a decentralized blockchain and a distributed content-addressable

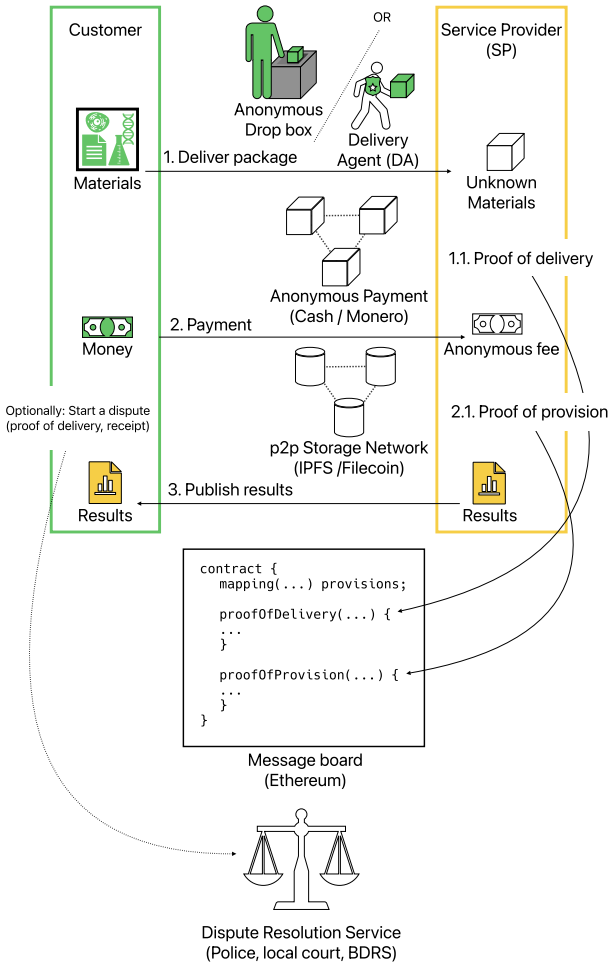

Fig. 1 A simplified diagram of the protocol. The first step involves the delivery of the materials to the SP. The second step involves anonymous payment. The third step involves the delivery of the result to the customer. Each step is proven on the message board, protecting the fair party in a conflict situation

storage network, while acknowledging the role of offline dispute resolution.
  - Enhances **user experience and practicality** in conflict-free transactions by minimizing customer interaction with the blockchain.
  - Ensures **remote availability of results**, even if the service provider becomes unresponsive.

- We provide a working prototype with open-source code, demonstrating the practical applicability of our protocol.
- We clarify key concepts such as anonymity, pseudonymity, linkability, and traceability, which are often used ambiguously in the literature.
- We introduce and validate a novel framework for analyzing fairness in fair exchange protocols.
- We discuss potential future enhancements using technologies like decentralized dispute resolution, self-sovereign identities, zero-knowledge proofs, and anonymous delivery.

Some authors have proposed blockchain-based fair exchange systems that could be adapted to service provision; however, to the best of our knowledge, we are the first to propose a system that satisfies all of the above properties. In particular, anonymity and physical delivery have rarely been addressed together, and if so, the protocol was based on TTP and impractical assumptions about the banking system [10] or did not address the conflict between parties [11].

The rest of this paper is organised as follows. In Section 2 we review related works. Then in Section 3 we discuss the building blocks of a dispute resolution system, blockchain as a message board, fairness, anonymous payments, storage network, availability of results and anonymity. Section 4 provides a detailed description of the protocol. Section 5 provides a fairness analysis of the proposed protocol. Section 6 presents the implementation of the protocol and the results of our experiments. Section 7 discusses possible improvements in terms of crowdsourced dispute resolution or dispute avoidance, self-sovereign identities (SSIs), anonymous delivery, and formal verification. Finally, Section 8 concludes the paper.

2 Related Works

This section reviews the main protocols for fair exchange, anonymity and physical delivery, highlighting their main features and differences.

The most common application of fair exchange protocols is in e-commerce. In a typical transaction, a seller and a customer exchange money for a physical product. To protect themselves, the seller wants to receive the funds before sending the product, while the customer wants to receive the product before paying. The fairness of the protocol should ensure that either both parties receive the goods or neither receives anything. Early systems, such as those by Zhang et al. (2006)[12] and Mohammedalaraj (2012)[13], introduced Trusted Third Parties (TTPs) and Delivery Agents (DAs) to ensure fairness and relied on strong assumptions about non-collusion and resilient communication channels.

Protocols such as those proposed by Bîrjoveanu et al. (2015-2022) [10, 14–17] focus on anonymity in transactions involving physical products. These protocols use TTPs, anonymous communication channels like Tor, and cryptographic techniques like blind signatures to ensure privacy and fairness. However, they rely on the existence of TTPs and secure, confidential transaction systems between banks.

Blockchain technology offers solutions to TTP dependency in fair exchange protocols. Hinarejos et al. (2019)[18] demonstrated a blockchain-based protocol for certified email

that replaces TTP with a decentralized, verifiable system. Themis (Meng et al., 2019)[19] took this further by incorporating a decentralized dispute resolution mechanism, although it does not address anonymity.

Lelantos (Altawy et al., 2017) [11] is a notable example of a blockchain-based system that provides anonymous physical delivery. It uses onion routing and smart contracts to ensure anonymity, although it only achieves pseudonymity and lacks a dispute resolution mechanism.

Comparison and our approach We have only considered protocols that achieve fair exchange, as this is the fundamental feature of such protocols. We also didn't focus on protocols for buying digital products, as they are not relevant to our use case. A more comprehensive analysis of such protocols is available in [17].

Altawy et al. 2017 [11] is a blockchain-based protocol that uses onion routing and anonymous blockchain interaction to provide anonymous physical delivery, assuming unlinkability between pseudonyms and real identities. However, it does not provide dispute resolution. Hinarejos et al. 2019 [18] is the simplest protocol that replaces TTP with blockchain. However, it does not take into account anonymity, disputes between parties, or the exchange of physical material. Meng et al. 2019 [19] improves on the previous protocol through a crowdsourced dispute resolution system. However, it does not consider anonymity. Bîrjoveanu, 2022 [17] is the closest to our protocol, but it is based on strong assumptions, namely the existence of TTP, banks supporting confidential transactions with commit buffers, and maintaining a global list of coin serial numbers.

Our protocol differs from existing ones in that it focuses on anonymity and fair exchange without relying on TTPs or complex banking systems. It achieves anonymity by using either cash or privacy-preserving blockchains. In addition, our protocol does not require the customer to submit a transaction to the bulletin board, simplifying the transaction process while maintaining anonymity and fairness.

Table 1 provides a summary of the key features and differences between the protocols discussed.

Table 1 Comparison of related works. Category Definitions: *E-commerce:* Secure online transactions for physical goods; *Private Delivery System:* Methods ensuring fair and anonymous delivery of physical products; *Certified email:* Secure exchange of certified electronic mail with proof of receipt; *Digital escrow:* Decentralized escrow services for secure digital currency transactions. The notation *Pseudonymity* means that the anonymity is based on the assumption that the pseudonym is not linked to the real identity; *TTP* means that the protocol relies on a trusted third party in every transaction; *Offline TTP* means that the protocol relies on a trusted third party only to resolve disputes; *BC* means that the protocol uses a public blockchain; *YES** means that the protocol provides the feature but is based on strong or impractical assumptions

Protocol	Scenario	Fair exchange	Anonymity	Dispute resolution	Trust	Physical delivery
Zhang et al. [12] (2006)	E-commerce	Yes	Yes*	Yes	TTP	Yes
Alaraj et al.[13] (2012)	E-commerce	Yes*	No	Yes	Offline TTP	Yes
Lelantos [11] (2017)	Private Delivery System	Yes	Pseudonym	No	BC	Yes
Hinarejos et al.[18] (2019)	Certified email	Yes	No	No	BC	No
Themis [19] (2019)	Digital escrow	Yes	No	Yes	BC	No
PPPDCP [17] (2022)	E-commerce	Yes	Yes	Yes	TTP	Yes
This paper	E-commerce	Yes	Yes	Yes	BC & Offline TTP	Yes

3 Building Blocks

3.1 Physical Products

In the realm of fair exchange protocols involving physical materials, existing solutions often rely on trusted intermediaries [10, 13] or complex delivery systems [11] to maintain anonymity. While these methods are effective, they can be cumbersome and less practical for certain use cases.

In our context, the process is reversed, as the physical materials are transferred from an anonymous customer to a publicly known service provider (SP). This unique

setup allows for a simplification of the delivery process. We propose several methods to facilitate anonymous delivery without compromising the customer's personal information:

- **SP's Drop Box:** Utilizing a secure drop box provided by the SP, where customers can leave their packages without revealing their identity.
- **Parcel Locker Services:** Leverage existing locker services (e.g. Amazon Locker, InPost) that provide a level of anonymity and security for package delivery.
- **Trusted Delivery Agent:** The customer can use a trusted individual or service to deliver the package to the SP, ensuring the customer's anonymity.
- **Postal Services:** Traditional postal services may be used, provided they do not require personal identification or return addresses that could compromise anonymity.

3.2 Dispute Resolution System

Disputes are common in transactions, and systems are needed to ensure that rules and laws are followed. Traditionally, this has involved legal contracts and law enforcement. With blockchain technology, smart contracts offer a new way by putting the details of a contract into code and enforcing it through a consensus mechanism. [20, 21].

Smart contracts often require integration with real-world data, which is facilitated by *oracles* that bridge off-chain information to the blockchain [22]. This is particularly relevant for decentralized dispute resolution systems like Kleros [23–25] or Themis [19]. These platforms utilize decentralized networks of jurors, who are often domain experts, to review smart contracts and evidence in case of disputes and deliver verdicts back to the blockchain. Such systems often involve mechanisms for staking and penalizing misbehavior to ensure fair arbitration. However, these systems may face challenges with complex disputes requiring highly specialized knowledge or handling sensitive, private information.

Our protocol adopts a more conventional approach, leveraging the blockchain for evidence logging but relying on established legal systems (police or courts) for dispute resolution. This approach balances technological innovation with the nuanced understanding required for complex service-related disputes. While this introduces a degree of centralization in dispute resolution, the blockchain infrastructure ensures transparency and immutability of evidence. We further explore integrating decentralized, partially automated dispute resolution in Section 7.2 as a potential enhancement towards greater decentralization.

3.3 Fairness

In our protocol, disputes are resolved by providing evidence to judicial authorities (police or courts). As the customer remains anonymous, the SP cannot initiate a dispute due to the inability to identify the customer. Our protocol is designed to favour the SP who adheres to the protocol, thus eliminating the need to initiate disputes. Conversely, customers can initiate disputes, but only misbehavior by the SP will result in a successful claim.

We outline three key pieces of evidence for dispute resolution:

1. **Proof of Delivery (PoD)**): Issued by the SP to confirm that the customer has delivered a complete package in accordance with the SP's requirements. The detailed definition is given in Section 4.2.
2. **Payment receipt**: This confirms that the customer has paid for the service. Its format varies depending on the payment method (cryptocurrency or cash) and is discussed further in Section 3.6.
3. **Proof of Provision (PoP):** This shows that the SP has published the service result at a certain time, thus protecting the SP against unjustified disputes from the customer. The detailed definition is given in Section 4.2.

3.4 Message Board

The platform for issuing and publishing these proofs is a critical aspect of the protocol. Known by various names such as bulletin board [26], trusted timestamping [27], or message board [18], this platform serves as a decentralized, public ledger that securely timestamps and records each proof. By using blockchain technology, the platform ensures that the proofs are tamper-proof and verifiable by any third party, including judicial authorities.

Using a decentralized message board offers several advantages:

- It provides an immutable record of interactions and agreements between the customer and service provider.
- It allows transparent verification of event timelines, critical for resolving disputes fairly.
- It offers a universally accessible platform to verify the authenticity and integrity of transaction proofs.

The protocol is designed to be adaptable, allowing for implementation with any decentralized technology capable of providing a message board service and supporting subscription to proofs from specific addresses.

The choice between permissioned and permissionless blockchains involves a trade-off between control and openness. While permissioned blockchains are suitable for spe-

cific domain applications, such as consortia of known participants, our protocol is designed for a broader, more open context. Existing fair exchange protocols often utilize public blockchains, highlighting their relevance to this problem. Public blockchains offer key benefits like decentralization, transparency, openness, and reduced infrastructure maintenance costs (transaction costs only). They also provide a robust security model, particularly in networks with a large number of validators. Furthermore, protocols designed for public blockchains are generally more adaptable and can be implemented on private or permissioned networks if needed. Therefore, we focus on public, permissionless networks for our protocol, recognizing that permissioned blockchains could be considered as alternative deployment environments for specific use cases.

3.5 Anonymity, Pseudonymity, and Confidentiality

Privacy, a multifaceted concept in social sciences, is often ambiguously defined [1]. In our context, we focus on more concrete aspects: confidentiality, anonymity and pseudonymity.

Confidentiality refers to the ability to conceal the details of actions within a system. A system guarantees confidentiality if observers can only ascertain that an action occurred, without additional information.

Anonymity involves hiding one's identity within a system. It's the inability to link actions to a user's identity. Anonymity is a spectrum, quantifiable by k-anonymity [28], where a user is k-anonymous if their actions are indistinguishable from k-1 other users. The larger the k, the greater the anonymity.

Pseudonymity differs from anonymity. Users operate under pseudonyms, and while actions can be linked to these pseudonyms, the system remains anonymous as long as the real identities behind these pseudonyms are concealed. However, this assumption is challenging due to KYC (Know Your Customer) and AML (Anti Money Laundering) regulations, which require users to reveal their real identities to cryptocurrency exchanges or other on-ramping services. This exposure of users' privacy to government agencies, malicious insiders, and cybercriminals complicates the maintenance of true anonymity and raises concerns about transaction analysis [29, 30].

Figure 2 illustrates these concepts. Alice, wanting anonymity, controls two addresses. The first address's link to her identity is compromised, but the second remains anonymous. Her actions from these addresses demonstrate the nuances of confidentiality and anonymity.

In the realm of blockchain technologies, the concepts of anonymity and confidentiality are achieved through different mechanisms, depending on whether the blockchain is privacy-preserving or not. We outline these differences below:

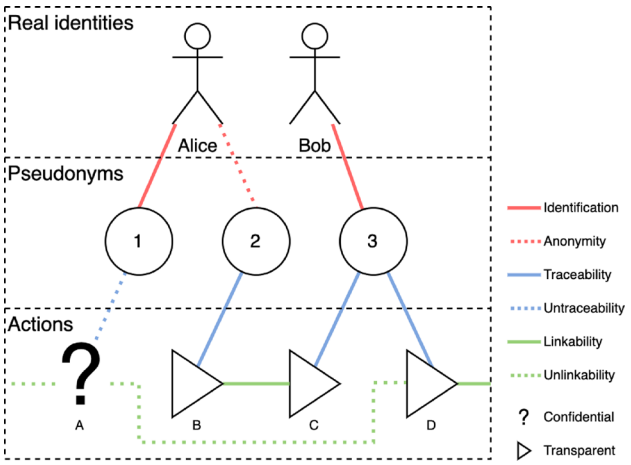

Fig. 2 Consider a scenario where Alice is an anonymous customer and Bob is a public service provider (SP). Alice controls two addresses, labelled 1 and 2. The link (shown as a red line) between Alice's real identity and address 1 has been compromised, allowing identification. However, the link to address 2 remains unknown, preserving anonymity. Alice performs two actions, A and B. Action A is from the compromised address 1, while action B is from the anonymous address 2. Action A is confidential, meaning that despite the compromised pseudonym, the nature of the action remains undisclosed. Action B is transparent, but it cannot be linked to Alice as long as the connection to pseudonym 2 remains hidden

1. **Privacy preserving blockchains:** These blockchains inherently support both anonymity and confidentiality through advanced cryptographic techniques, ensuring that both user identities and transaction details are obscured. Examples include **Monero**, which uses ring signatures and bulletproofs [31–33], **ZCash**, which uses zkSNARKs for private transactions [34], and **Grin** and **IronFish**, which use Mimblewimble and the Sapling protocol respectively [35–38].
2. **Non-privacy preserving blockchains:** These blockchains, such as Bitcoin and Ethereum, do not natively support strong privacy features. However, anonymity can be enhanced through the use of mixers and other privacy tools. Examples include **Tornado Cash** (Ethereum), which implements zkSNARKs and MiMC for enhanced privacy [39, 40], and **Wasabi Wallet** (Bitcoin), which uses CoinJoin to mix transactions [41, 42].

3.6 Paying for services

In transactions between customers and service providers (SPs), it is essential to establish a unique link between the payment and the corresponding transaction in order to prevent the reuse of a payment for multiple transactions. This link can be established in various ways depending on the cryptocurrency used:

– **Separate address:** Each transaction uses a unique address associated with it. These addresses can be generated using Hierarchical Deterministic Wallets [43] and published on the message board to ensure non-repudiation.
– **Memo:** Payments are sent to a single SP account but contain an extra field called "memo" filled with the unique identifier provisionID.

Any payment that contains provisionID in the memo or is sent to the specified address will be recognized as payment for the transaction.

In the event of a dispute, it is necessary to prove to the courts that the customer has paid for the transaction. While proving payment is straightforward in transparent and traceable blockchains, it becomes more complex in anonymous blockchains. Monero, for example, allows payments to be proven and verified through a dedicated API [44], and ZCash provides a mechanism known as Payment Disclosure [45]. We refer to this proof of payment as a *payment receipt*.

3.7 Storage Network

Upon completion of the service, the Service Provider (SP) faces the challenge of delivering the result to the customer while maintaining the customer's anonymity. The customer who wishes to remain anonymous cannot reveal their email address or IP address. In addition, the SP must prove that the result was delivered before a specified deadline, according to the concept of proof of existence discussed in section 3.4.

Even though the result does not contain private information (because the input data did not have it) we acknowledge the danger of storing on a distributed immutable record as blockchain.

A common solution is to use a content addressable peer-to-peer storage network for data storage. This approach has been widely adopted in various applications [46–48]. Specifically, data is stored on a network such as IPFS [49], and only the content identifier (cid) that uniquely points to the content on IPFS is published on the blockchain.

Following this methodology, the SP encrypts the result using the encryption key provided by the client and uploads it to the IPFS network.

The result is stored on public network so it is accessible to anyone with the cid address. However, the result is encrypted, so even if downloaded by third party it doesn't get access to the file unless it obtains the client's or SP's private key.

To further increase anonymity, customers are advised to use standard techniques such as VPNs or proxies to hide their IP addresses.

3.8 Provable Results Availability

The demand-driven and opportunistic nature of IPFS storage means that results may not be available indefinitely. To ensure the availability of the results, we see two approaches. One involves fraud proofing, where a user who cannot access the results calls an Oracle service such as Chainlink Request [22, 50], which proves the unavailability of the content and thus pentalises the SP.

The other approach is to integrate Filecoin as a decentralised pinning service to ensure the results are available for a defined period of time. Filecoin increases the availability of content on the IPFS network by economically penalising the lack of proof of content storage [51]. The availability of content is therefore economically guaranteed for the duration of the Filecoin contract. A file that is no longer paid for in the Filecoin network will naturally be deleted from the network, as there is no incentive to keep it.

In this paper we explore the second approach, leaving the first for future work.

3.9 Separation of Concerns

The protocol we propose could potentially use a single blockchain to fulfil three different roles: i) facilitating anonymous payments, ii) serving as a message board, and iii) acting as a storage network. However, while most blockchains can provide message board functionality, anonymous payments and a verifiable storage network are less common features and are often limited to specialized blockchains.

Rather than relying on a single blockchain to provide all functionalities, our protocol is designed to be flexible, allowing the use of separate blockchains for each specific role. This approach provides the freedom to choose the best available technology for each function. In the future, if a blockchain emerges that can efficiently handle multiple roles, it can be integrated into the protocol.

Based on the current state of blockchain technology, we identify the following platforms as suitable candidates for each role:

1. **Anonymous payments:** Technologies such as Monero [31], ZCash [34], Grin [36] and Tornado Cash [40] provide robust solutions for anonymous transactions.
2. **Message Board:** Several platforms can be used for this purpose, including Open Timestamps [52], Stampery [53] and the Bitcoin blockchain with services like Proof of Existence [54] and Chainpoint [55]. The Ethereum blockchain and other public blockchains that support attaching extra data to transactions are also viable options.

3. **Storage Network:** For decentralized storage solutions, IPFS [49], Filecoin [56] and Ethereum's Swarm [57] are among the leading technologies.

4 The Protocol

This section outlines an abstract protocol for anonymous service provisioning that is designed to be technology agnostic. It specifies the requirements for each role, allowing developers flexibility in technology choice. Implementation details and experimental validation of this protocol are discussed in Section 6.

4.1 Assumptions

The protocol is based on several key assumptions, each critical to its functionality and security:

- **Cryptography and Public Key Infrastructure (PKI)**:
  - The service provider (SP) has a key pair consisting of a secret key (sk_{SP}) and a publicly known public key (pk_{SP}).
  - Digital signatures created by the SP ($sig_{sk_{SP}}$) can be verified using the public pk_{SP}.
  - The customer remains anonymous and does not require a publicly known key pair, thus ensuring their privacy and anonymity in the protocol.
  - Both parties use standard symmetric encryption ($E_{key}(\cdot)$) and decryption ($D_{key}(\cdot)$) methods.
- **Service Provider (SP) Requirements**:
  - The SP is willing to accept packages from unidentified customers.
  - Payments are accepted in cash or anonymous cryptocurrencies, as detailed in Section 3.6.
- **Anonymous Payments Blockchain**:
  - Facilitates anonymous transactions that are untraceable and ideally unlinkable.
  - Allows transactions to be uniquely identified through dedicated addresses, memo fields or similar mechanisms (see Section 3.6).
- **Message Board**:
  - Capable of handling transaction sizes up to PoD and PoP.
- **Storage Network**:
  - Enables content to be retrieved using a content identifier (cid), typically a hash of the content.
  - Ensures anonymous access to content.
  - Guarantees that the content will be available for an agreed period of time.
- **Dispute Resolution Service**:
  - Recognizes PoD (Proof of Delivery), PoP (Proof of Provision), and payment receipts as valid evidence in disputes (see Section 3.3).

4.2 Messages

This subsection details the messages exchanged between the parties in the protocol.

Package is a physical container prepared by the customer containing all the materials required by the SP to provide the service.

$$\text{pkg} \equiv (\text{materials}, \text{provisionID}, \text{pk}_C) \tag{1}$$

where:

- materials - Materials required for the service provision, such as biological samples, legal documents or other relevant items.
- provisionID - A unique identifier generated by the customer to anonymously track the provision through all protocol steps.
- pk_C - The customer's public key for encrypting results to be published to the public storage network.

Proof of Delivery (PoD) confirms the correct delivery of the package to the SP and its acceptance.

It is also an agreement between the customer and the SP as it contains the description of the expected item [1], agreed deadlines for actions and a payment method.

$$\begin{aligned}\text{PoD} \equiv (&\text{pk}_C, \text{provisionID}, \text{itemDescription},\\ &\text{paymentAddress}, T_{issue}, T_{pay}, T_{provide}, \text{sig}_{SP})\end{aligned} \tag{2}$$

where:

- pk_C - Customer's public key.
- provisionID - Unique transaction identifier.
- itemDescription - Hash of the description of the expected item.
- paymentAddress - SP's anonymous payment address.
- T_{issue} - Time of PoD issuance.
- T_{pay} - Payment deadline.
- $T_{provide}$ - Service provision deadline.
- sig_{SP} - SP's digital signature.

[1] We follow the assumption from [58] that there exists a string that describes the desired item in "sufficient" detail so that during the dispute the arbitrator can decide whether the provided result is as agreed.

also: $T_{issue} \leq T_{pay} \leq T_{provide}$

Proof of Provision (PoP) verifies the SP's publication of results at a specific time, protecting the SP against unjustified disputes.

The link between PoP and the results is made by the content identifier (cid), which uniquely identifies the results so that the result cannot be forged after the PoP has been published.

$$\text{PoP} \equiv (\text{cid}, \text{provisionID}, \text{sig}_{SP}) \tag{3}$$

where:

- cid - Content identifier as specified in Section 3.7.
- provisionID - Unique transaction identifier.
- sig_{SP} - SP's digital signature.

Payment-receipt proves that the customer made the payment. Since the proof depends on a specific blockchain (see Section 3.6), we symbolically refer to it as receipt.

Results are typically in PDF format, but any binary-encodable format acceptable for the storage network can be used, symbolically referred to as results.

Content Identifier (cid) is a term from IPFS [59]. However, the protocol allows for any secure and unique identifier for content referencing.

4.3 Protocol Description

This section outlines each step of the protocol, also shown in Figure 3.

Step 0. Preparation

The customer prepares the materials required by the SP, generates a random provisionID, and a keypair (sk_C, pk_C). The provisionID and pk_C are encoded as a QR code, attached to the pkg. The sk_C is kept secret for decrypting the result later.

Step 1. Package Delivery

The protocol initiates when the customer delivers pkg to the SP. The SP creates PoD with deadlines T_{pay}, $T_{provide}$, and T_{issue}. The PoD also specifies the description of the item $itemDescription$, and payment method (cash or blockchain address). The SP's digital signature $\text{sig}_{sk_{SP}}$ on PoD ensures non-repudiation.

Symbolically:

$$\text{PoD} \leftarrow \text{delivery}(\text{pkg})$$

Step 2. Proof of Delivery

The SP publishes PoD on the message board, confirming receipt of pkg. If not paid in cash, the SP awaits payment at the specified address in PoD.

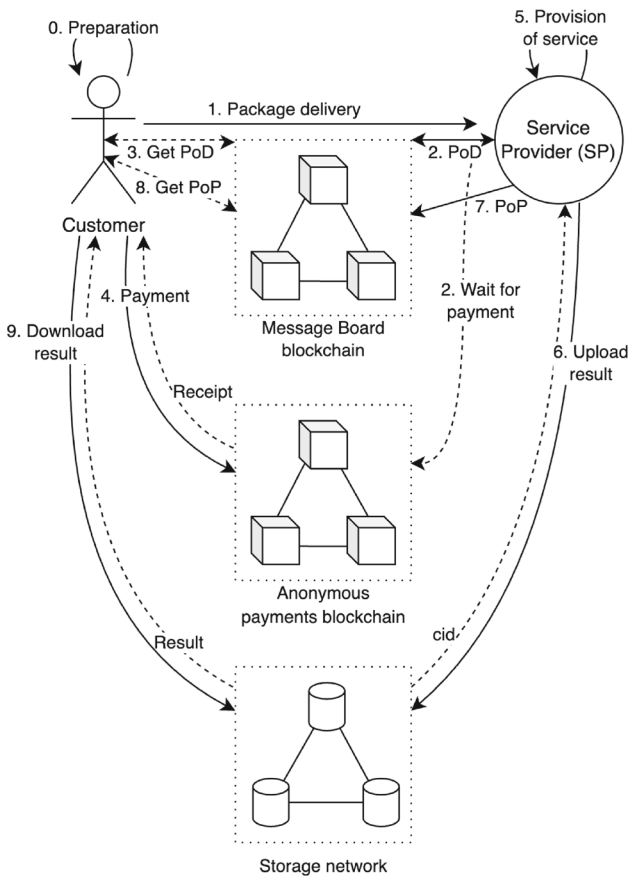

Fig. 3 Messages exchanged in the protocol. Solid arrows indicate requests and dashed arrows indicate responses

Symbolically:

$$\text{publish}(\text{PoD})$$

The Algorithm 1 presents the process of processing the PoD and recording it on-chain.

Step 3. Get Proof of Delivery

The customer retrieves PoD from the message board and verifies payment details, $itemDescription$ and time windows $(T_{issue}, T_{pay}, T_{provide})$.

Symbolically:

$$\text{PoD} \leftarrow \text{get}(pk_C, \text{provisionID})$$

Step 4. Payment

If not paid in cash, the customer pays via the anonymous blockchain, receiving a receipt.

Symbolically:

$$\text{receipt} \leftarrow \text{payment}(\text{paymentAddress})$$

Step 5. Provision of Service

Algorithm 1 Algorithm for Registering Proof of Delivery

```
function ProofOfDelivery(pk_C, provisionId, itemDescription, paidInCash, paymentAddress, paymentWindow, provisionWindow, sig_SP)
    Verify sig_SP
    if provisions[pk_C][provisionId] is not empty then
        return Error("Provision already exists")
    end if
    if paidInCash = TRUE ⊕ paymentAddress is not empty then
        return Error("Pay in cash or specify paymentAddress")
    end if
    T_issue ← block.timestamp
    T_pay ← T_issue + paymentWindow
    T_provide ← T_issue + provisionWindow
    provision ← {T_issue, T_pay, T_provide
        itemDescription, paidInCash, paymentAddress}
    provisions[pk_C][provisionId] ← provision
end function
```

Upon payment confirmation, the SP begins service provision.

Symbolically:

result ← provision(materials)

Step 6. Upload Result

The SP encrypts the result using $E_{DHKE(sk_{SP},pk_C)}$ and uploads it to the storage network, receiving a cid.

Symbolically:

cid ← upload($E_{DHKE(sk_{SP},pk_C)}$(result))

Step 7. Proof of Provision

The SP publishes PoP on the message board, including cid and provisionID.

Symbolically:

publish(PoP)

The Algorithm 2 presents the process of processing the PoP and recording it on-chain.

Algorithm 2 Algorithm for Registering Proof of Provision

```
function ProofOfProvision(pk_C, provisionId, cid, sig_SP)
    Verify sig_SP
    provision ← provisions[pk_C][provisionId]
    if not provision.exist then
        return Error("Provision must be created with proof of delivery first")
    end if
    Update provision with {cid}
    provisions[pk_C][provisionId] ← provision
end function
```

Step 8. Get Proof of Provision

The customer monitors the message board for the SP's PoP.

Symbolically:

cid ← get(pk_C, provisionID)

Step 9. Download Result

The customer downloads and decrypts the result using $D_{DHKE(sk_C,pk_{SP})}$.

Symbolically:

result ← $D_{DHKE(sk_C,pk_{SP})}$(download(cid))

5 Fair Exchange Analysis

Fair exchange protocols involve two parties trading items with each other, where each party holds an item to trade and expects to receive a specific item in return. These protocols operate even when the parties do not necessarily trust each other. A key goal of fair exchange protocols is to ensure that a dishonest participant cannot exploit the situation to gain an advantage over an honest participant. Specifically, these protocols must meet the following requirements [58, 60]:

1. **R1, Fairness**: Fairness can be defined in two ways:
   - **R1a Strong fairness**: Upon protocol completion, either both participants receive their desired items (successful exchange) or neither does (exchange fails).
   - **R1b Weak fairness**: When strong fairness is not achievable, an honest participant can demonstrate to an external arbiter that the other party has received (or still can receive) the expected item.
2. **R2, Timeliness**: An honest participant can be assured that the exchange will conclude (either successfully or unsuccessfully) within a specified timeframe, regardless of the other participant's behavior. At the end of the protocol, the outcome is final from the honest participant's perspective (e.g., the fairness ensured by the protocol remains consistent).
3. **R3, Effectiveness**: The protocol guarantees successful completion of the exchange if both participants act correctly and agree to trade.

5.1 Weak fairness

Here we demonstrate that with the use of an external arbitrator (Dispute Resolution Service), our protocol achieves **R1b Weak Fairness**: either both parties receive their desired items at the end of the protocol, or neither receives anything. Specifically, the customer receives the service result, and the

SP receives payment, or neither occurs. This is in contrast to **R1a Strong Fairness**, which would ideally guarantee simultaneous exchange without reliance on an external arbiter. Achieving strong fairness in scenarios involving physical goods and anonymous transactions is inherently challenging, which is why we focus on the more practical and commonly achievable weak fairness.

First, we model the protocol as an interactive non-cooperative game to analyze strategic interactions between the service provider (SP) and the customer. We then show how successful protocol executions lead to both parties obtaining their desired outcomes, while unsuccessful executions result in neither party gaining an advantage. We also incorporate penalties for misbehavior in our model to discourage unwarranted disputes.

5.1.1 Model

We define three distinct positions to represent the state of each party within the protocol:

- **Neutral Position** (•): The state of a party when no significant resources (money, time, effort) have been expended or acquired.
- **Disadvantaged Position** (−): The state of a party when it has invested resources without receiving a commensurate return, like a customer who has paid in advance for a service.
- **Advantaged position** (+): A scenario where stopping the transaction would result in a benefit to one party, e.g. where the SP has been paid but has not yet delivered the service.

The actions of each party are categorized as follows:

1. **Normal** (n): Those who adhere to the prescribed steps of the protocol.
2. **Abnormal** ($\overline{\text{n}}$): Any deviation from the protocol's prescribed steps, such as sending irrelevant messages, skipping steps, or exceeding time limits.

Additionally, the customer has the option to initiate a dispute at any protocol step, introducing another strategic layer:

1. **Agree** (n or $\overline{\text{n}}$): The customer consents to the action and refrains from disputing.
2. **Start a dispute** (d or $\overline{\text{d}}$): The customer opposes the action and initiates a dispute.

Consequently, our analysis needs to consider four possible action $\in \{\text{n}, \overline{\text{n}}, \text{d}, \overline{\text{d}}\}$, for each party $\in \{\text{C}, \text{SP}\}$, at every protocol step step $\in 1..9$:

1. $\sigma_{\text{step,party,n}}$: The outcome after adhering to the protocol with the other party acting normally.
2. $\sigma_{\text{step,party,d}}$: The outcome following a resolved dispute with the other party acting normally.
3. $\sigma_{\text{step,party},\overline{\text{n}}}$: The outcome when no dispute is raised despite the other party's abnormal actions.
4. $\sigma_{\text{step,party},\overline{\text{d}}}$: The outcome after a resolved dispute with the other party acting abnormally.

The protocol ends after the final step, if a dispute is initiated, or if a party fails to take the required action within the specified timeframe. Therefore, all positions except $\sigma_{\text{step,party,n}}$ for step $\in 1..8$ indicate the end of the protocol.

The anonymity of the customer prevents the SP from initiating a dispute. To address this, the protocol is designed in such a way that an SP who adheres to the protocol remains in an advantageous position, eliminating the incentive to dispute. Conversely, the customer can dispute at any time, but only proven misbehavior of the SP will result in a successful dispute. The Algorithm 3 presents the process of dispute resolution.

Algorithm 3 Algorithm for Dispute Resolution

function DISPUTE RESOLUTION(pk_C, $provisionId$, $receipt$, $result$)
  $provision \leftarrow provisions[pk_C][provisionId]$
  if Time dispute **then**
    Verify $provision.paidInCash$ OR (valid $receipt$ AND $receipt$.time < $provision.\text{T}_{\text{pay}}$)
    Verify not $provision.cid$ and $provision.provisionTime > provision.\text{T}_{\text{provide}}$
    return The Customer wins the dispute because the SP has not provided the PoP on the agreed time.
  end if
  if Result dispute **then**
    $provision.cid \equiv result$
    $provision.itemDescription \not\equiv result$
    return The Customer wins the dispute because the SP has not provided the desired result.
  end if
end function

5.1.2 Assumptions

For the purpose of our analysis, we operate under the following assumptions:

1. Both parties start from a neutral position (•), implying no initial advantage or disadvantage.
2. Upon successful completion of the transaction, both parties reach an advantageous position (+), indicating mutual benefit and motivation to initiate and complete the transaction.

3. The steps within the protocol are atomic, i.e. they are indivisible and have no intermediate states.
4. The protocol is unidirectional; actions taken cannot be reversed or undone.
5. The protocol can only be restarted by repeating the first step. Any repetition of subsequent steps is considered abnormal and will be disregarded. For example, a double payment does not change the course of the protocol.
6. Once the result of the service is published, it becomes available to the customer via the storage network.
7. A successful dispute resolution restores the disputing party to a neutral position (•).
8. Losing a dispute incurs a penalty that exceeds any potential gain, resulting in a disadvantaged position (-). This discourages frivolous or uncertain disputes.
9. Both the customer and the SP are rational actors who prefer to move from a less advantageous to a more advantageous position. However, they may temporarily accept a less advantageous position if it leads to a subsequent advantageous state, as long as there is an escape route from the less advantageous position. Specifically, the client can initiate a dispute to move from a disadvantaged (-) to a neutral (•) position if the SP fails to comply with the protocol.
10. The customer's materials, without personal information, have no value and the effort to deliver the package is considered minimal. Thus, the customer's first step does not lead to a disadvantaged position.
11. The cost of publishing the Proof of Delivery (PoD) is negligible and is offset by the customer's effort in delivering the package.

5.2 Proofs

Theorem 1 *The proposed protocol satisfies the security requirement of* ***R1b Weak fairness****.*

Proof The positions of each party after each action within our protocol are visually represented in Figure 4. The detailed description of each step and the reasoning behind the outcomes is given in Appendix A.

Following the definition of **R1b Weak fairness** it's sufficient to show that, with the use of an external arbitrator (Dispute Resolution Service), the protocol completes either with both parties getting the desired items (advantaged positions), or neither does (neutral or disadvantaged). The protocol (successfully) completes at the last normal agreement step ($\sigma_{9,n}$), and (unsuccessfully), after starting a dispute (σ_{d} or $\sigma_{\overline{\mathrm{d}}}$). We consider only honest (rational) customer, and so it will always starts a dispute after the SP misbehaves, and so it will never settle at abnormal agreement positions ($\sigma_{\overline{\mathrm{n}}}$, as that would be irrational. □

Step	Position / Turn	Normal				Abnormal			
		Agreemet		Dispute		Agreemet		Dispute	
		C	SP	C	SP	C	SP	C	SP
1	Customer	•	•	–	•				
2	Service Provider	•	•	–	•	•	•	–	•
3	Customer	•	•	–	•	•	•	–	•
4		–	+	–	•	•	•	–	•
5	Service Provider	–	+	–	•	–	+	•	–
6		–	+	–	•	–	+	•	–
7		–	+	–	•	–	+	•	–
8	Customer	–	+	–	•	–	+	–	•
9		+	+	–	•	–	+	–	•

Fig. 4 Visual representation of the fairness of the protocol. This figure shows the outcomes for each party after various actions. The symbols used are • for a neutral outcome, - for a disadvantaged outcome, and + for an advantaged outcome. The order relation is defined as - < • < +

Theorem 2 *The proposed protocol satisfies the security requirement of* ***R2, Timeliness****.*

Proof The timeliness of the protocol is achieved through the use of blockchain. Assuming an honest majority, the blockchain acts as a global, immutable and undeniable clock that timestamps every action and moves the protocol forward. Every step must be recorded on the blockchain, once the transaction is recorded it's undeniable.

Both parties agree to the timelines $\mathrm{T}_{\mathrm{issue}}$, $\mathrm{T}_{\mathrm{pay}}$, $\mathrm{T}_{\mathrm{provide}}$ and so any breach of the timelines is undeniably conclusive. The blockchain continuously creates blocks at a probabilistic rate. If a party hasn't completed its step within a time window, it can't publish a transaction in a previous block. Consequently, the protocol cannot move from a completed to an uncompleted state, nor can it hang in an uncompleted position, as the blocks produced move the parties' states out of the agreed timelines, thus fulfilling the **R2, Timeliness** requirement. □

Theorem 3 *The proposed protocol satisfies the security requirement of* ***R2, Effectiveness****.*

The first scenario (Figure 5) represents the rational and intended course of action, where both the client and the SP follow the protocol as designed. This scenario is crucial as it shows that parties who act correctly (normally and agreeably) will eventually succeed and thus **R3, Effectiveness** is achieved.

Step	Position / Turn	Normal				Abnormal			
		Agreemet		Dispute		Agreemet		Dispute	
		C	SP	C	SP	C	SP	C	SP
1	Customer	•	•	–	•				
2	Service Provider	•	•	–	•	•	•	–	•
3	Customer	•	•	–	•	•	•	–	•
4		–	+	–	•	•	•	–	•
5	Service Provider	–	+	–	•	–	+	•	–
6		–	+	–	•	–	+	•	–
7		–	+	–	•	–	+	•	–
8	Customer	–	+	–	•	–	+	–	•
9		+	+	–	•	–	+	–	•

Fig. 5 Transitions of positions in a scenario where both the customer and the SP adhere to the protocol

Step	Position / Turn	Normal				Abnormal			
		Agreemet		Dispute		Agreemet		Dispute	
		C	SP	C	SP	C	SP	C	SP
1	Customer	•	•	–	•				
2	Service Provider	•	•	–	•	•	•	–	•
3	Customer	•	•	–	•	•	•	–	•
4		–	+	–	•	•	•	–	•
5	Service Provider	–	+	–	•	–	+	•	–
6		–	+	–	•	–	+	•	–
7		–	+	–	•	–	+	•	–
8	Customer	–	+	–	•	–	+	–	•
9		+	+	–	•	–	+	–	•

Fig. 6 Transitions of positions in a scenario where the SP misbehaves by failing to perform the service and publish PoP after receiving payment. This leads to the customer initiating a dispute

5.3 Dispute scenario

In the second scenario (Figure 6), (1) the customer initialises the transaction and delivers the material to the SP, (2) who validates the material and publishes the Proof of Delivery (PoD) on blockchain. (3) The customer validates the existence of the PoD on the blockchain, and (4) pays for the transaction using one of the specified payment methods. (5) The SP misbehaves by not provisioning the service and consequently not publishing the Proof of Provision (PoP) after receiving the payment. Once the agreed $T_{provide}$ deadline has passed, the position moves to the Abnormal column and the customer is in a disadvantaged position and the SP is in an advantaged position. However, the customer is justified in starting a dispute by collecting all the evidence, i.e. Proof of Delivery (PoD) from the message board, payment recipt, and submitting it to the dispute resolution service. The customer is likely to win this dispute as the SP cannot prove the timely publication of PoP. This scenario results in a neutral outcome for the customer and a disadvantaged outcome for the SP.

6 Experiments

The prototype of our protocol has been developed using a number of technologies designed to ensure anonymity and security:

- **Anonymous Payments:** Use of the Monero blockchain for secure transactions.
- **Storage Network:** Powergate serves as an interface for Filecoin and IPFS, facilitating decentralized storage.
- **Message Board:** The Ethereum blockchain, accessed via a local development version Truffle Ganache and Solidity, acts as a public ledger.
- **Customer and SP Interface:** A client-side web application built with `React.js` and `web3.js`, with `MetaMask` for Ethereum transactions and `monero-wallet-cli` for Monero interactions.

We used the following tools to create the experiment: `monerod` and `monero-wallet-cli` - v0.18.1.2; `Powergate` - v2.6.2; `Ganache` - v7.5.0; `Solidity` - v0.8.17; `ReactJS` - v18.0.25; `web3.js` - v1.8.1; `crypto-js` - v4.1.1.

For simplicity, all components run on one physical machine; and all processes are managed by Docker. Moreover, Powergate is configured to use local Filecoin and IPFS networks. For the Ethereum blockchain, we use Truffle Ganache, which is a local Ethereum blockchain for development and testing purposes. Monero is configured to use the public stage network. We assume that the service provider offers only one type of service at a fixed public price, so we omit the service type and price from the protocol.

The source code is available at https://github.com/stanbar/anonser.

In preparation for the experiment, both the customer and the SP set up their Monero wallets using the `monero-wallet-cli`. The SP deploys the smart contract to the Ethereum blockchain with the `truffle migrate --network development` command, and the web application is then configured to interact with this newly deployed

contract. The customer acquires test Monero funds from a Faucet service and configures their wallet to generate transaction proofs, essential for dispute resolution. With preparations complete, the experiment proceeds as follows:

0. **Setup:** The customer begins by creating a new provision in the webapp, which generates a unique ECDSA keypair and provisionID. The corresponding QR code is printed and attached to the parcel for delivery.
1. **Package Delivery:** The parcel is delivered to the SP through a chosen delivery method, such as a secure drop box or locker service. Upon receipt, the SP scans the QR code to retrieve the provisionID and customer's public key. Since (in this experiment) the provision was not paid in cash, the SP generates a unique Monero payment address using `monero-wallet-cli integrated_address`. As a result, the PoD is prepared.
2. **Proof of Delivery:** Then, the SP submits the PoD to the Ethereum blockchain using the MetaMask interface.
3. **Get Proof of Delivery:** The customer checks the transaction status on the Ethereum blockchain by calling `getProvision` with arguments `customerPubKey` and `provisionID`
4. **Payment:** The customer sends the payment (using `monero-wallet-cli transfer`) to the `paymentAddress` specified in the smart contract and stores the payment receipt using `monero-wallet-cli get_tex_key < tx-id >`.
5. **Provision of Service:** Upon payment confirmation, the SP provides the service, resulting in a `result.pdf` file.
6. **Upload result:** This file is uploaded to IPFS and Filecoin, granting the SP a `cid`, `dealID`, and `minerID`.
7. **Proof of Provision:** A PoP is submitted to the Ethereum blockchain by the SP.
8. **Get Proof of Provision:** Meanwhile, the customer subscribes to Ethereum and waits for the SP to publish the PoP.
9. **Download result:** Upon noticing the PoP, the customer retrieves the result using either one of the public gateways[2] like https://cf-ipfs.com/ipfs/<cid> or Lotus network using `lotus retrieve <cid> <minerID>`. The result is then decrypted using the customer's previously stored private key. If the customer is satisfied with the service the protocol ends; otherwise, the customer may initiate a dispute process.

[2] IPFS Public Gateway Checker, https://ipfs.github.io/public-gateway-checker/, (last visited Jan. 04, 2023)

6.1 Results

Fairness As shown in Section 5.2 the protocol is fair. This was achieved through an undeniable handshake mechanism, where the SP first commits to the package delivery and service deadlines by publishing the PoD (as outlined in step 2). The customer then acknowledges this commitment and accepts the terms by proceeding with the payment for the service (step 3).

Once payment is confirmed, the SP is incentivized to fulfill the service obligations. The SP must deliver the service and publish both the results and the PoP before the agreed deadline (step 7). Failure to do so allows the customer, equipped with all necessary evidence, to initiate a dispute and potentially penalize the SP. This mechanism ensures that rational parties are motivated to adhere to the protocol.

The protocol also ensures non-repudiation without the need for a TTP by employing blockchain technology and digital signatures. The blockchain provides a transparent and immutable record, ensuring that any changes to the smart contract's state are publicly visible and can only be made by the SP.

Anonymity Anonymity was ensured by breaking the link between personal data and transactional elements, including materials, payments, and communications. We used anonymous payment methods such as cash and privacy-centric cryptocurrencies like Monero, to conceal transactional details. Furthermore, we leveraged decentralized storage networks like IPFS and Filecoin, which facilitate the anonymous storage and retrieval of data. This approach guaranteed that, in the dispute-free transaction, customer interactions with the protocol remained confidential at every stage.

Depending on the specific use case, the anonymity of the customer may be lost if the resolution of the dispute (e.g. by the police) involves the identification of the customer. However, even if the Dispute Resolution Service identifies the customer, the customer's identity won't be disclosed to the SP, which is the main goal of the protocol.

To maintain anonymity in the event of a dispute, Online Dispute Resolution systems [20, 21] such as Kleros [23–25] should be used, where only the customer's pseudonym is revealed and anonymous evidence can be provided in zero-knowledge proofs. We discuss this further in Section 7.2.

Provable Results Availability The availability of the result is guaranteed by the usage of Filecoin [56], which operates as an incentivization layer on top of IPFS. Filecoin enhances content availability by economically penalizing the lack of proof of content storage [51].

In our protocol, the SP is responsible for uploading the result to both the IPFS and Filecoin networks (utilizing Powergate), which ensures free access to the results under normal operational circumstances. This dual-network approach also

Table 2 Incurred Costs for Protocol Operations

Operation	Gas Units	Cost (USD)
Smart Contract Deployment	1,456,577	4.09
Proof of Delivery	129,649	0.29
Payment (Monero)	-	0.0456
Proof of Provision	149,130	0.33

ensures high availability of the results, even if the SP ceases to host the content on their node.

Costs The deployment and operation of smart contracts on the Ethereum blockchain incur gas fees, which are proportional to the computational resources required for transaction execution. The following outlines the gas consumption and associated costs for each operation within our protocol, based on the testnet metrics, which are analogous to the mainnet:

The cost of gas is denominated in ETH, and the price per unit of gas at the time of the experiment (January 3, 2023) was 0.000000002227 ETH/gas, with the ETH price being $1,261.97 USD[3]. The customer is responsible only for covering the payment transaction fee. For transactions using Monero, the fee was approx. 0.000304 XMR at the time of the experiment, with the price of Monero being $150 USD per XMR[4].

The incurred costs are summarized in Table 2:

Additionally, our protocol's interaction with the Filecoin network introduces costs in FIL cryptocurrency for data storage and retrieval. These costs are determined through market-driven deals with miners. Deal prices, quoted in FIL, are influenced by various factors including data size, storage duration, and miner policies. The dynamic nature of these parameters means that costs can fluctuate, making precise predictions challenging. However, for our experiment, we leveraged Filecoin's reputation-based incentivisation layer to publish deals at no cost[5].

Performance Evaluation Metrics To provide a comprehensive performance evaluation of our protocol, we have analyzed the following metrics for each of the blockchain networks, as summarized in Table 3:

These metrics are crucial for understanding the scalability and efficiency of the blockchain networks in question and provide insight into their suitability for various applications within our protocol.

[3] Etherscan Gas Tracker, https://etherscan.io/gastracker

[4] Cryptocurrency statistics, https://bitinfocharts.com

[5] Filecoin, Filecoin Plus Overview, https://docs.filecoin.io/store/filecoin-plus/overview/

Table 3 Performance Evaluation Metrics

Network	Block Time	Tx Throughput	Tx Latency
Ethereum	13-15 sec	15-30 TPS	~6 min
Monero	~2 min	~4 TPS	~20 min
Filecoin	30 sec	N/A[6]	5-10 min

[6] The network is optimized for storage operations rather than transaction processing, with the throughput being primarily dependent on the storage and retrieval deal proposals

7 Discussion

7.1 Scalability and Practicality Considerations

For our protocol to transition from demonstration to real-world impact, careful consideration must be given to both scalability and practicality, particularly regarding high-demand applications and ease of use for a wider audience. Currently, the reliance on on-chain transactions for proofs introduces potential bottlenecks in scalability, with latency and costs that could become significant in large-scale deployments, especially on blockchains like Ethereum. Furthermore, the protocol's design assumes a degree of technical expertise from users, requiring them to manage cryptographic elements and engage with blockchain technologies, which presents a barrier to adoption for non-technical individuals, potentially limiting its accessibility in sectors like healthcare and legal services.

Future development should prioritize addressing these practical aspects. From a scalability standpoint, exploring Layer-2 solutions for Ethereum or migrating to alternative blockchains with higher throughput and lower transaction fees are viable paths. Simultaneously, enhancing practicality necessitates a focus on user experience. Creating intuitive interfaces, developing user-friendly applications, and potentially incorporating intermediary services to abstract away technical complexities will be crucial to broaden accessibility. Addressing both scalability and user experience in future work will be essential to realize the full potential of our anonymous and fair service provision protocol and facilitate its widespread adoption.

7.2 Dispute Resolution Service

A key challenge in developing fully decentralized Web3 systems is addressing the potential centralization of dispute resolution services [61]. To move towards greater decentralization in this aspect, we consider the integration of blockchain-based dispute resolution systems.

Blockchain-based dispute resolution platforms, such as Kleros [23–25], Themis [19], Aragon Court [62, 63], LTO Network [64], and other Online Dispute Resolution (ODR)

systems [20, 21], offer a promising avenue for future development. These systems could utilize decentralized panels of field experts who would review smart contracts, proofs (PoD, PoP, payment receipt), and other evidence to reach verdicts in disputes. Mechanisms involving fees, stakes, and rewards are typically used to incentivize expert participation and ensure impartial judgments. Kleros, a prominent platform in this domain, has demonstrated success in resolving consumer disputes in e-commerce and the collaborative economy, and is actively supported by the European Commission [65]. With a track record of resolving thousands of disputes and employing a large pool of jurors [66], Kleros represents a viable option for integration with protocols like ours, which we plan to explore in future work.

7.3 Security Considerations and Future Formal Verification

While our protocol incorporates cryptographic techniques and blockchain technology to enhance security, a comprehensive security analysis is essential.

One consideration is the reliance on IPFS and Filecoin for result availability. While Filecoin incentivizes storage and enhances availability, the long-term persistence of data depends on the economic incentives and continued operation of the Filecoin network. If these incentives were to fail, or if miners were to cease storing the data, result availability could be compromised. Smart contract vulnerabilities are another potential concern. Although Solidity and Ethereum are mature technologies, vulnerabilities in smart contracts can still arise, potentially leading to exploits. Rigorous smart contract auditing and formal verification are crucial to mitigate these risks. Furthermore, while our protocol focuses on anonymity, metadata leaks or sophisticated traffic analysis could potentially lead to de-anonymization in certain scenarios, especially if users do not employ best practices like VPNs or Tor.

To enhance the security analysis of our protocol, future work should include formal verification using tools like AVISPA [67], following methodologies similar to [68]. Formal verification would involve specifying the protocol in a language like HLPSL [69] and using automated tools to rigorously check for security properties. This would provide a more robust and mathematically grounded assurance of the protocol's security.

7.4 Self-sovereign Identities

In our exploration of privacy-preserving protocols, we encountered regulatory requirements that mandate the linking of diagnostic results to patient identities, as seen in regulations like those in Poland [70]. Such regulations pose a direct challenge to protocols designed for strict anonymity.

Self-sovereign identities (SSIs) and verifiable claims offer a potential approach to reconcile anonymity with certain regulatory needs [71]. In an SSI framework, a trusted authority, such as a government agency, could issue a one-time verifiable claim to a customer. Service providers could accept this claim as a form of pseudo-identification, linking diagnostic results to a Decentralized Identifier (DID) that itself contains no personal data. While the SP would not be able to directly identify the customer, the issuing authority could, if legally necessary, trace the DID back to the individual.

Given that SSI technology is still in its early stages and lacks widespread governmental adoption, further research and development are necessary to fully assess its applicability and effectiveness in balancing anonymity and regulatory compliance in sensitive service provision.

7.5 Ethical and Regulatory Considerations

Our protocol aims to enable anonymous access to services, which inherently raises ethical and regulatory considerations, particularly concerning Know Your Customer (KYC), Anti-Money Laundering (AML), and regulations related to Politically Exposed Persons (PEP).

Our protocol is designed to be integrated with services that operate within existing legal and regulatory frameworks. While the protocol itself prioritizes anonymity, it does not aim to facilitate illegal activities. The types of services envisioned for our protocol are generally not those directly involved in financial transactions or money transmission that typically fall under stringent KYC/AML requirements. However, we acknowledge the concerns around anonymous payment methods like Monero and their potential association with illicit activities. It is important to emphasize that Monero is just one of several payment options supported by our protocol. The protocol is designed to be flexible and can accommodate various payment mechanisms, including those that incorporate KYC/AML compliance at the service provider level if required by applicable regulations or service policies.

For instance, while we discuss the use of privacy mixers to enhance payment anonymity on platforms like Ethereum, we recognize the regulatory scrutiny these technologies face due to potential misuse. Recent sanctions against services like Tornado Cash highlight these concerns. Ongoing research into compliant privacy mixers, such as Voluntary Reveal Approaches or retroactive de-anonymization [72] with sanctioned list (like Chainalysis [73]), offers potential paths towards more regulator-friendly anonymous payment solutions. These approaches often involve mechanisms for providing proofs of innocence or integrating with oracles that monitor sanctioned addresses, potentially enabling a balance between privacy and compliance. Ultimately, the choice of payment method and the level of anonymity employed

would need to be carefully considered in the context of specific service applications and relevant legal and regulatory requirements.

7.6 Permissioned Blockchain Alternatives

While our protocol is designed for permissionless public blockchains to maximize openness, transparency, and censorship resistance, we acknowledge that permissioned blockchain solutions offer a different set of trade-offs that may be advantageous in certain contexts. In scenarios where regulatory compliance, auditability, and strict access control are paramount, particularly in highly regulated sectors like healthcare or finance, a permissioned blockchain infrastructure might be more suitable. Permissioned blockchains allow for controlled participation, enabling authorized entities like healthcare providers, laboratories, and patients to operate within a closed and auditable system. Access control mechanisms can be implemented to ensure that sensitive data, such as medical records, are only accessible to authorized personnel. Furthermore, integrating KYC-compliant payment methods and identity verification within a permissioned blockchain framework can directly address AML and regulatory concerns for high-value transactions. Therefore, while our work focuses on the benefits of permissionless systems for anonymous service provision, we recognize the viability and potential advantages of permissioned blockchain alternatives for specific use cases where different priorities and regulatory requirements prevail.

8 Conclusions

This study has been dedicated to the development of a protocol that facilitates the provision of services while preserving the anonymity of the user. Our protocol is particularly applicable to services requiring a high degree of confidentiality, such as genetic testing, paternity determination, and anonymous legal consultation.

We found that the current state of the art was not sufficient to achieve this goal, so we have designed and implemented a novel protocol that ensures user anonymity, fairness in service delivery, and a mechanism for dispute resolution without the need for a trusted third party. This protocol leverages anonymous payment systems, such as cash or privacy-focused cryptocurrencies, and utilizes peer-to-peer networks for the dissemination of service results.

Through rigorous definition and analysis, we have demonstrated that our protocol meets the criteria for a fair exchange protocol, as outlined in Section 5. It does this by systematically publishing proofs of delivery, payment and provisioning, ensuring a transparent and fair process for all parties involved.

In closing, we have pinpointed several avenues for future enhancement, including the integration of decentralized dispute resolution systems, the application of self-sovereign identity (SSI) frameworks, and the exploration of anonymous physical delivery methods. These areas present exciting opportunities for further research and development towards the realization of fully anonymous service provision in the digital age.

Appendix A Proof of fairness

Below we describe each step and the reasoning behind the outcome position. We use the notation introduced in Section 5.1.1 to analyse each position in the protocol and to show the fairness of the protocol.

Step 1. Customer turn: Package delivery

The protocol starts when the customer correctly completes the first step of the protocol, i.e. delivers the package to the SP.

The case where the customer does not deliver the package is not considered as it is not part of the protocol.

- Agreeable path:
  - $\sigma_{1,c,n} = \bullet$, the customer risked his materials but did not pay for the transaction and therefore ends up in a neutral position (see Assumption 10. in Section 5.1.2).
  - $\sigma_{1,s,n} = \bullet$, the SP ends up in a neutral position as she did not spend any resources and the package did not bring her any value.
- The customer starts a dispute:
  - $\sigma_{1,c,d} =$ -, The customer loses the dispute because the SP is not obliged to do anything until the transaction is paid.
  - $\sigma_{1,s,d} = \bullet$, The SP wins the dispute for the same reason.

Fairness:

- The customer can follow the protocol to the non-disadvantaged position $\sigma_{1,c,n} = \bullet$
- The SP can do nothing and always ends up in the non-disadvantaged position

Step 2. SP turn: Proof of Delivery

The SP publishes the PoD, then:

- Agreeable path:
  - $\sigma_{2,c,n} = \bullet$, the customer remains in the neutral position as the PoD allows him to pay for the transaction but does not oblige him to do anything.
  - $\sigma_{2,s,n} = \bullet$, the SP remains in the neutral position as the package has not brought her any value and she has not spent any resources to provide the service.
- The customer starts a dispute:
  - $\sigma_{2,c,d} =$ -, The customer loses the dispute because the SP is not obliged to do anything until the transaction is paid.
  - $\sigma_{2,s,d} = \bullet$, The SP wins the dispute for the same reason.

The SP acted abnormally, then:

- Agreeable path:
  - $\sigma_{2,c,\bar{n}} = \bullet$, the customer remains in the neutral position as he is not obliged[6] to agree with the incorrect PoD.
  - $\sigma_{2,s,\bar{n}} = \bullet$, the SP remains in the neutral position as the package has not brought her any value and she has not spent any resources to provide the service.
- The customer starts a dispute:
  - $\sigma_{2,c,\bar{d}} =$ -, The customer loses the dispute because the SP is not obliged to do anything until the transaction is paid, not even to publish correct PoD.
  - $\sigma_{2,s,\bar{d}} = \bullet$, The SP wins the dispute for the same reason.

Fairness:

- The SP can do anything and always ends up in the non-disadvantaged position.
- customer can either wait (if the SP is following the protocol) or abandon the transaction (if the SP is acting abnormally). In both cases the customer ends up in a non-disadvantaged position $\sigma_{2,c,n} = \bullet$ or $\sigma_{2,c,\bar{n}} = \bullet$.

Step 3. Customer turn: Get Proof of Delivery

The customer got the PoD, then:

- Agreeable path:
  - $\sigma_{3,c,n} = \bullet$, The customer remains in the neutral position.
  - $\sigma_{3,s,n} = \bullet$, The SP remains in the neutral position.
- The customer starts a dispute:
  - $\sigma_{3,c,d} =$ -, The customer loses the dispute because the SP is not obliged to do anything until the transaction is paid.
  - $\sigma_{3,s,d} = \bullet$, The SP wins the dispute for the same reason.

The Customer acted abnormally, then:

- Agreeable path:
  - $\sigma_{3,c,\bar{n}} = \bullet$, The customer remains in the neutral position.
  - $\sigma_{3,s,\bar{n}} = \bullet$, The SP remains in the neutral position.
- The customer starts a dispute:
  - $\sigma_{3,c,\bar{d}} =$ -, The customer loses the dispute because the SP is not obliged to do anything until the transaction is paid.
  - $\sigma_{3,s,\bar{d}} = \bullet$, The SP wins the dispute for the same reason.

Fairness:

- The customer can follow the protocol to the non-disadvantaged position $\sigma_{3,c,n} = \bullet$.
- The SP can do nothing and always ends up in the non-disadvantaged position.

Step 4. Customer turn: Payment

The customer paid the transaction, then:

- Agreeable path:
  - $\sigma_{4,c,n} =$ -, the customer has paid in advance.
  - $\sigma_{4,s,n} = +$, the SP has received the payment but has not spent his resources yet.
- The customer starts a dispute:
  - $\sigma_{4,c,d} =$ -, The customer loses the dispute because the SP is still able to publish the PoP within the agreed timeframe.
  - $\sigma_{4,s,d} = \bullet$, The SP wins the dispute for the same reason.

The Customer acted abnormally, then:

[6] By not obliged we understand the situation where a party does not risk any resources by not taking the action

- Agreeable path:
  - $\sigma_{4,c,\bar{n}} = \bullet$, the customer ends up in the neutral position as he has not spent his funds.
  - $\sigma_{4,s,\bar{n}} = \bullet$, the SP ends up in the neutral position as she neither received the payment nor spent her resources.
- The customer starts a dispute:
  - $\sigma_{4,c,\bar{d}} =$ -, The customer loses the dispute because the SP is not obliged to do anything until the transaction is paid.
  - $\sigma_{4,s,\bar{d}} = \bullet$, The SP wins the dispute for the same reason.

Fairness:

- The customer, following the 9th assumption described in Section 5.1.2, risks the temporary disadvantaged position $\sigma_{4,c,n} =$ - in favour of a later better position $\sigma_{9,s,n} = +$; in the meantime, he can get out of the disadvantaged position if the SP misbehaves in any of the following steps.
- The SP can do nothing and always ends up in the non-disadvantaged position.

Step 5. SP turn: Provision of service

The SP did the provision of service, then:

- Agreeable path:
  - $\sigma_{5,c,n} =$ -, The Customer remains in the disadvantaged position as he hasn't received the result.
  - $\sigma_{5,s,n} = +$, The SP remains in the advantaged position as she has received the payment.
- The customer starts a dispute:
  - $\sigma_{5,c,d} =$ -, The customer loses the dispute because the SP is still able to publish the PoP within the agreed timeframe.
  - $\sigma_{5,s,d} = \bullet$, The SP wins the dispute for the same reason.

The SP acted abnormally, then:

- Agreeable path:
  - $\sigma_{5,c,\bar{n}} =$ -, The customer ends up in a disadvantageous position, because he has paid in advance, but hasn't received the result.
  - $\sigma_{5,s,\bar{n}} = +$, The SP ends up in the advantageous position, having received the payment.
- The customer starts a dispute:
  - $\sigma_{5,c,\bar{d}} = \bullet$ The customer wins the dispute because the SP has not provided the service within the time agreed in the PoD, and therefore the SP is unable to upload the result and publish the PoP on time.
  - $\sigma_{5,s,\bar{d}} =$ -, The SP loses the dispute for the same reason.

Fairness:

- The customer, following the 9th assumption described in Section 5.1.2, risks the temporary disadvantaged position $\sigma_{5,c,n} =$ - in favour of a later better position $\sigma_{9,s,n} = +$; in the meantime, he can get out of the disadvantaged position if the SP misbehaves in any of the following steps.
- The SP can follow the protocol and move to the advantaged position $\sigma_{5,s,n} = +$, or act abnormally (not provide the service) and also move to the advantaged position $\sigma_{5,s,\bar{n}} = +$; however, the second option puts her at risk of terminating the protocol at $\sigma_{5,s,\bar{d}} =$ - if the customer is rational and starts a dispute; hence, the SP should choose the first option.

Step 6. SP turn: Upload result

The SP uploaded the result on time, then:

- Agreeable path:
  - $\sigma_{6,c,n} =$ -, The Customer remains in the disadvantaged position as he has not received the result.
  - $\sigma_{6,s,n} = +$, The SP remains in the advantaged position as she has received the payment.
- The customer starts a dispute:
  - $\sigma_{6,c,d} =$ -, The customer loses the dispute because the SP is still able to publish the PoP within the agreed timeframe.
  - $\sigma_{6,s,d} = \bullet$, The SP wins the dispute for the same reason.

The SP acted abnormally, then:

- Agreeable path:
  - $\sigma_{6,c,\bar{n}} =$ -, The customer ends up in a disadvantageous position, because he has paid in advance, but hasn't received the result.
  - $\sigma_{6,s,\bar{n}} = +$, The SP ends up in the advantageous position, having received the payment.
- The customer starts a dispute:

  - $\sigma_{6,c,\bar{d}} = \bullet$, the customer wins the dispute because the SP has not uploaded the service within the time agreed in the PoD and the SP will not be able to publish the PoP on time. $\sigma_{6,s,\bar{d}} =$ -, The SP loses the dispute for the same reason.

Fairness:

- The customer, following the 9th assumption described in Section 5.1.2, risks the temporary disadvantaged position $\sigma_{6,c,n} =$ - in favour of a later better position $\sigma_{9,s,n} = +$; in the meantime, he can get out of the disadvantaged position if the SP misbehaves in any of the following steps.
- The SP can follow the protocol and move to the advantaged position $\sigma_{6,s,n} = +$, or act abnormally (not provide the service) and also move to the advantaged position $\sigma_{6,s,\bar{n}} = +$; however, the second option puts her at risk of terminating the protocol at $\sigma_{6,s,\bar{d}} =$ - if the customer is rational and starts a dispute; hence, the SP should choose the first option.

Step 7. SP turn: Proof of provision

The SP published PoP on time, then:

- Agreeable path:
  - $\sigma_{7,c,n} =$ -, the customer has not received the result. Therefore, he remains in a disadvantaged position.
  - $\sigma_{7,s,n} = +$, the SP has published all the evidence to prove her correct behaviour, so she remains in an advantageous position for the rest of the protocol.
- The customer starts a dispute:
  - $\sigma_{7,c,d} =$ -, the customer loses the dispute as the SP has published all evidences to prove her correct behaviour.
  - $\sigma_{7,s,d} = +$, The SP wins the dispute for the same reason.

The SP acted abnormally, then:

- Agreeable path:
  - $\sigma_{7,c,\bar{n}} =$ -, The customer ends up in a disadvantageous position, because he has paid in advance, but hasn't received the result.
  - $\sigma_{7,s,\bar{n}} = +$, The SP ends up in the advantageous position, having received the payment.
- The customer starts a dispute:
  - $\sigma_{7,c,\bar{d}} = \bullet$, the customer wins the dispute because the SP did not publish the correct PoP on time.
  - $\sigma_{7,s,\bar{d}} =$ -, The SP loses the dispute for the same reason.

Fairness:

- The customer, following the 9th assumption described in Section 5.1.2, risks the temporary disadvantaged position $\sigma_{7,c,n} =$ - in favour of a later better position $\sigma_{9,s,n} = +$; in the meantime, he can get out of the disadvantaged position if the SP misbehaves in any of the following steps.
- The SP can follow the protocol and move to the advantaged position $\sigma_{7,s,n} = +$, or act abnormally (not provide the service) and also move to the advantaged position $\sigma_{7,s,\bar{n}} = +$; however, the second option puts her at risk of terminating the protocol at $\sigma_{7,s,\bar{d}} =$ - if the customer is rational and starts a dispute; hence, the SP should choose the first option.

Step 8. Customer turn: Get Proof of Provision

The customer got the PoP, then:

- Agreeable path:
  - $\sigma_{8,c,n} =$ -, the customer gets the cid, but not the result yet.
  - $\sigma_{8,s,n} = +$, The SP remains in the advantaged position.
- The customer starts a dispute:
  - $\sigma_{8,c,d} =$ -, the customer loses the dispute as the SP has published all evidences to prove her correct behaviour.
  - $\sigma_{8,s,d} = +$, The SP wins the dispute for the same reason.

The Customer acted abnormally, then:

- Agreeable path:
  - $\sigma_{8,c,\bar{n}} =$ -, the customer has paid for the transaction but does not have access to the cid and therefore cannot get the result from the storage network.
  - $\sigma_{8,s,\bar{n}} = +$, The SP ends up in the advantageous position, having received the payment.
- The customer starts a dispute:
  - $\sigma_{8,c,\bar{d}} =$ -, the customer loses the dispute as the SP has published all evidences to prove her correct behaviour

  - $\sigma_{8,s,\bar{d}} = \bullet$, The SP wins the dispute for the same reason.

Fairness:

- The customer, following the 9th assumption described in Section 5.1.2, risks the temporary disadvantaged position $\sigma_{8,c,n} =$ - in favour of a later better position $\sigma_{9,s,n} = +$; in the meantime, he can get out of the disadvantaged position if the SP misbehaves in any of the following steps.
- The can do nothing and always ends up in the non-disadvantaged position $\sigma_{8,s,n} = +$ or $\sigma_{8,s,\bar{n}} = +$.

Step 9. Customer turn: Download result

The customer downloaded the result, then:

- Agreeable path:
  - $\sigma_{9,c,n} = +$, The customer has received the result, therefore he finishes the protocol in an advantaged position.
  - $\sigma_{9,s,n} = +$, The SP remains in the advantaged position.
- The customer starts a dispute:
  - $\sigma_{9,c,d} =$ -, the customer loses the dispute as the SP has published all evidences to prove her correct behaviour.
  - $\sigma_{9,s,d} = +$, The SP wins the dispute for the same reason.

The Customer acted abnormally, then:

- Agreeable path:
  - $\sigma_{9,c,\bar{n}} =$ -, the customer ends up in a disadvantaged position, as he ends up with the incorrect result.
  - $\sigma_{9,s,\bar{n}} = +$, the SP ends up in the advantageous position of having received the payment but not having spent his resources.
- The customer starts a dispute:
  - $\sigma_{9,c,\bar{d}} = \bullet$, the customer wins the case and ends up in the neutral position.
  - $\sigma_{9,s,\bar{d}} =$ -, the SP loses the case and ends up in the disadvantaged position.

Fairness:

- The Customer can follow the protocol to the non-disadvantaged position $\sigma_{9,c,n} = +$.
- The can do nothing and always ends up in the non-disadvantaged position $\sigma_{9,s,n} = +$ or $\sigma_{9,s,\bar{n}} = +$.

Acknowledgements The research was supported partially by the project "Cloud Artificial Intelligence Service Engineering (CAISE) platform to create universal and smart services for various application areas", No. KPOD.05.10-IW.10-0005/24, as part of the European IPCEI-CIS program, financed by NRRP (National Recovery and Resilience Plan).

Author Contributions Stanislaw Baranski: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization Julian Szymanski: Supervision, Project administration, Funding acquisition, Validation, Writing - Original Draft, Writing - Review & Editing Higinio Mora: Validation Writing - Original Draft, Writing - Review & Editing, Methodology

Data availability No data was used for the research described in the article. The source code is available at https://github.com/stanbar/anonser.

Declarations

Conflict of interest The authors declare that they have no competing interests. The authors certify that they have no affiliations with or involvement in any organisation or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors

References

1. Smith, H Jeff, Dinev, Tamara, Xu, Heng: Information Privacy Research: An Interdisciplinary Review. MIS Quarterly **35**(4), 989–1015 (2011)
2. Klitzman, Robert: Exclusion of Genetic Information From the Medical Record. JAMA:the journal of the American Medical Association **304**(10), 1120–1121 (2010)
3. Black, Kevin J., Barton, Stacey K., Perlmutter, Joel S.: Presymptomatic Testing and Confidentiality in the Age of the Electronic Medical Record. The Journal of Neuropsychiatry and Clinical Neurosciences **33**(1), 80–83 (2021)
4. Jin, Hao, Luo, Yan, Li, Peilong, Mathew, Jomol: A review of secure and privacy-preserving medical data sharing. IEEE Access **7**, 61656–61669 (2019)

5. Keshta, Ismail, Odeh, Ammar: Security and privacy of electronic health records: Concerns and challenges. Egyptian Informatics Journal **22**(2), 177–183 (2021)
6. Naresh, Vankamamidi S., Thamarai, Muthusamy: Privacy-preserving data mining and machine learning in healthcare: Applications, challenges, and solutions. WIREs Data Mining and Knowledge Discovery **13**(2), e1490 (2023)
7. Chun-Wei Lin, Jerry, Fournier-Viger, Philippe, Wu, Lintai, Gan, Wensheng, Djenouri, Youcef, Zhang, Ji: PPSF: An open-source privacy-preserving and security mining framework. In 2018 IEEE International Conference on Data Mining Workshops (ICDMW), pages 1459–1463. IEEE, (2018)
8. Hewage, U.H.W.A., Sinha, R., Naeem, M Asif: Privacy-preserving data (stream) mining techniques and their impact on data mining accuracy: A systematic literature review. Artificial Intelligence Review **56**(9), 10427–10464 (2023)
9. Solove, Daniel J.: A taxonomy of privacy. University of Pennsylvania law review, pages 477–564, (2006)
10. Bîrjoveanu, Cătălin V.: Anonymity and fair-exchange in e-commerce protocol for physical products delivery. In 2015 12th International Joint Conference on E-Business and Telecommunications (ICETE), volume 04, pages 170–177, (July 2015)
11. AlTawy, Riham, ElSheikh, Muhammad, Youssef, Amr M., Gong, Guang: Lelantos: A Blockchain-Based Anonymous Physical Delivery System. In 2017 15th Annual Conference on Privacy, Security and Trust (PST), pages 15–1509, (August 2017)
12. Zhang, Q., Markantonakis, K., Mayes, K.: A Practical Fair-Exchange E-Payment Protocol for Anonymous Purchase and Physical Delivery. In IEEE International Conference on Computer Systems and Applications **2006**, 851–858 (2006)
13. Alaraj, Abdullah Mohammed: Fairness in Physical Products Delivery Protocol. International journal of Computer Networks & Communications **4**(6), 99–110 (2012)
14. Bîrjoveanu, Catalin V., Bîrjoveanu, Mirela: Preserving Anonymity in Fair Exchange Complex Transactions E-Commerce Protocol for B2C/B2B Applications. In: Proceedings of the 15th International Joint Conference on E-Business and Telecommunications, ICETE 2018 - Volume 1: DCNET, ICE-B, OPTICS, SIGMAP and WINSYS, Porto, Portugal, July 26-28, 2018., pages 265–276, (2018)
15. Bîrjoveanu, Cătălin V., Bîrjoveanu, Mirela: Anonymity in Complex Transactions for e-Business. In Mohammad S. Obaidat, editor, E-Business and Telecommunications, Communications in Computer and Information Science, pages 24–45, Cham, (2019). Springer International Publishing
16. Bîrjoveanu, Catalin V., Bîrjoveanu, Mirela: Fair Exchange E-Commerce Protocol for Multi-Chained Complex Transactions. In Proceedings of the 17th International Joint Conference on E-Business and Telecommunications, ICETE 2020 - Volume 3: ICE-B, Lieusaint, Paris, France, July 8-10, 2020., pages 49–60, (2020)
17. Cătălin, V.: Bîrjoveanu and Mirela Bîrjoveanu. Two-Party E-Commerce Protocols. In: Bîrjoveanu, C.ătălin V., Bîrjoveanu, Mirela (eds.) Secure Multi-Party E-Commerce Protocols. SpringerBriefs in Computer Science, pp. 15–42. Springer International Publishing, Cham (2022)
18. Hinarejos, M Francisca, Ferrer-Gomila, Josep-Lluis., Huguet-Rotger, Llorenc: A solution for secure certified electronic mail using blockchain as a secure message board. IEEE Access **7**, 31330–31341 (2019)
19. Meng, Hongwei, Bian, Evan, Tang, Cong: Themis: Towards decentralized escrow of cryptocurrencies without trusted third parties. In: 2019 Sixth International Conference on Software Defined Systems (SDS), pages 266–271. IEEE, (2019)
20. Allen, Darcy, Lane, Aaron, Poblet, Marta: The Governance of Blockchain Dispute Resolution. SSRN Electronic Journal, (2019)
21. Lingwall, Jeff, Mogallapu, Ramya: Should code be law? Smart contracts, Blockchain, and Boilerplate. University of Missouri-Kansas City Law Review **88**, 285 (2019)
22. Breidenbach, Lorenz, Cachin, Christian, Chan, Benedict, Coventry, Alex, Ellis, Steve, Juels, Ari, Koushanfar, Farinaz, Miller, Andrew, Magauran, Brendan, Moroz, Daniel: Chainlink 2.0: Next steps in the evolution of decentralized oracle networks. Chainlink Labs, (2021)
23. Bergolla, Luis, Seif, Karen, Eken, Can: Kleros: A socio-legal case study of decentralized justice & blockchain arbitration. Ohio St. J. on Disp. Resol. **37**, 55 (2022)
24. Nappert, Sophie, Ast, Federico: Decentralized justice: Reinventing arbitration for the digital age. Global Arbitration Rev, (2020)
25. Gudkov, Aleksei: Crowd arbitration: Blockchain dispute resolution. Legal Issues in the Digital Age, (2020)
26. Achenbach, Dirk, Kempka, Carmen, Löwe, Bernhard, Müller-Quade, Jörn.: Improved Coercion-Resistant electronic elections through deniable Re-Voting. USENIX Journal of Election Technology and Systems (JETS) **3**, 26–45 (2015)
27. Gipp, Bela, Meuschke, Norman, Gernandt, André: Decentralized Trusted Timestamping using the Crypto Currency Bitcoin, (2015)
28. Sweeney, Latanya: K-anonymity: A model for protecting privacy. International journal of uncertainty, fuzziness and knowledge-based systems **10**(05), 557–570 (2002)
29. Androulaki, Elli, Karame, Ghassan O., Roeschlin, Marc, Scherer, Tobias, Capkun, Srdjan: Evaluating User Privacy in Bitcoin. In Ahmad-Reza Sadeghi, editor, Financial Cryptography and Data Security, pages 34–51, Berlin, Heidelberg, (2013). Springer Berlin Heidelberg
30. Ober, Micha, Katzenbeisser, Stefan, Hamacher, Kay: Structure and anonymity of the bitcoin transaction graph. Future internet **5**(2), 237–250 (2013)
31. Van Saberhagen, Nicolas: CryptoNote v 2.0, (2013)
32. Noether, Shen: Ring Signature confidential transactions for monero. IACR Cryptology ePrint Archive **2015**, 1098 (2015)
33. Bünz, Benedikt, Bootle, Jonathan, Boneh, Dan, Poelstra, Andrew, Wuille, Pieter, Maxwell, Greg: Bulletproofs: Short Proofs for Confidential Transactions and More. In: 2018 IEEE Symposium on Security and Privacy (SP), pages 315–334, (May 2018)
34. Ben-Sasson, Eli, Chiesa, Alessandro, Garman, Christina, Green, Matthew, Miers, Ian, Tromer, Eran, Virza, Madars: Zerocash: Decentralized Anonymous Payments from Bitcoin. In 2014 IEEE Symposium on Security and Privacy, pages 459–474, (May 2014)
35. Jedusor, Tom Elvis: MIMBLEWIMBLE, (2016)
36. Fuchsbauer, Georg, Orrù, Michele, Seurin, Yannick: Aggregate cash systems: A cryptographic investigation of mimblewimble. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques, pages 657–689. Springer, (2019)
37. Hopwood, Daira, Bowe, Sean, Hornby, Taylor, Wilcox, Nathan: Zcash Sapling protocol specification, (2022)
38. Iron Fish. Private, anonymous, and easy to use cryptocurrency, (2023)
39. Groth, Jens: On the Size of Pairing-based Non-interactive Arguments. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques, pages 305–326. Springer, (2016)
40. Pertsev, Alexey, Semenov, Roman, Storm, Roman: Tornado Cash Privacy Solution Version **1**, 4 (2019)
41. Maxwell, Gregory: CoinJoin: Bitcoin privacy for the real world, (2013)
42. Wallet, Wasabi: Bitcoin privacy wallet with built-in coinjoin, (2023)
43. Wuille, Pieter: BIP32 - Hierarchical Deterministic Wallets, (February 2012)
44. Monero. How to prove payment, (2022)

45. Davies, Gareth: An Introduction to Payment Disclosure in Zcash, (December 2017)
46. Shahid, Affaf, Almogren, Ahmad, Javaid, Nadeem, Al-Zahrani, Fahad Ahmad, Zuair, Mansour, Alam, Masoom: Blockchain-Based Agri-Food Supply Chain: A Complete Solution. IEEE Access **8**, 69230–69243 (2020)
47. Wang, Shangping, Tang, Xixi, Zhang, Yaling, Chen, Juanjuan: Auditable Protocols for Fair Payment and Physical Asset Delivery Based on Smart Contracts. IEEE Access **7**, 109439–109453 (2019)
48. Chen, Yongle, Li, Hui, Li, Kejiao, Zhang, Jiyang: An improved P2P file system scheme based on IPFS and Blockchain. In 2017 IEEE International Conference on Big Data (Big Data), pages 2652–2657, (December 2017)
49. Juan Benet. IPFS - content addressed, versioned, P2P file system. CoRR, arXiv:1407.3561, (2014)
50. Chainlink. Chainlink: Make a GET Request, (2024)
51. Filecoin. Slashing, (2023)
52. OpenTimestamps. A timestamping proof standard, (2023)
53. de Pedro Crespo, Adán Sánchez, García, Luis Iván Cuende: Stampery Blockchain Timestamping Architecture (BTA) - Version 6, (2017)
54. Proof of Existence. A web application to prove the existence of documents using the blockchain, (2023)
55. Chainpoint. Blockchain Proof & Anchoring Standard, (2023)
56. Protocol Labs. Filecoin: A Decentralized Storage Network, (2017)
57. Swarm Team. SWARM: Storage and Communication Infrastructure for a Self-Sovereign Digital Society, (2021)
58. Asokan, Nadarajah: Fairness in Electronic Commerce. PhD thesis, University of Waterloo, (1998)
59. IPFS. Content Identifiers (CIDs) | IPFS Docs, (2023)
60. Liu, Jian, Li, Wenting, Karame, Ghassan O., Asokan, N.: Toward Fairness of Cryptocurrency Payments. IEEE Security & Privacy **16**(3), 81–89 (2018)
61. Ethereum. What is Web3 and why is it important?, (March 2023)
62. Aragon. Aragon/whitepaper, (2023)
63. Aragon. Decentralized Dispute Resolution Protocol, (2023)
64. Network, L.T.O.: Next-Gen blockchain for B2B, Identities. Ownership and Digital Collectibles, Privacy (2023)
65. European Commission. The Commission's European Innovation Council awards € 5 million to blockchain solutions for social innovations | Shaping Europe's digital future, (2020)
66. Kleros. Kleros Homepage, (2024)
67. Armando, Alessandro, Basin, David, Boichut, Yohan, Chevalier, Yannick, Compagna, Luca, Cuellar, Jorge, Drielsma, Paul, Héam, Pierre-Cyrille, Kouchnarenko, Olga, Mantovani, Jacopo, Mödersheim, Sebastian , von Oheimb, David, Rusinowitch, Michael , Santiago, Judson, Turuani, Mathieu, Vigano, Luca, Vigneron, Laurent: The AVISPA tool for the automated validation of internet security protocols and applications. In Computer Aided Verification, volume 3576 of Lecture Notes in Computer Science, pages 135–165, Berlin, Heidelberg, (July 2005). Springer
68. Bîrjoveanu, Cătălin V., Bîrjoveanu, Mirela: Formal Verification of Multi-party Fair Exchange E-Commerce Protocols, pages 81–106. Springer International Publishing, Cham, (2022)
69. Chevalier, Yannick, Compagna, Luca, Cuellar, Jorge, Drielsma, Paul Hankes, Mantovani, Jacopo, Mödersheim, Sebastian, Vigneron, Laurent: A high level protocol specification language for industrial security-sensitive protocols. In Workshop on Specification and Automated Processing of Security Requirements-SAPS'2004, pages 13–p. Austrian Computer Society, (2004)
70. Zdrowia, Ministerstwo: Regulation of the Minister of Health of March 23, 2006, on quality standards for medical diagnostic and microbiological laboratories, (April 2006)
71. Mühle, Alexander, Grüner, Andreas, Gayvoronskaya, Tatiana, Meinel, Christoph: A survey on essential components of a self-sovereign identity. Computer Science Review **30**, 80–86 (2018)
72. Baranski, Stanislaw, Dotan, Maya, Lotem, Ayelet, Vald, Margarita: Haze and Daze: Compliant Privacy Mixers
73. Chainalysis. Chainalysis oracle for sanctions screening, 2024