

Universidade do Minho
Escola de Engenharia

Data Security

Group Project

pg59788 - José Diogo Azevedo Martins

pg59789 - Luis Enrique Diaz De Freitas

pg59802 - Tomás Moura Martins dos Santos Ferreira

Index

1. Introduction	1
2. Requirements & Threat Model	1
2.1. System Requirements	1
2.2. Threat Model	1
3. System Architecture	2
3.1. Components	2
4. Identity & Anonymity Framework	3
4.1. Dual-Identity Model	4
4.2. Blind-Signed Anonymous Tokens	4
4.3. Encrypted Identity Package (Revocable Anonymity)	5
5. Auction Protocol & Lifecycle	5
5.1. Auction Creation	6
5.2. Bid Message Format	6
5.3. Validation Rules	6
5.4. Winner Determination & Identity Reveal	7
5.4.1. Winner Determination & Identity Reveal	7
5.4.2. Mutual Authentication (Seller's Reveal)	7
5.4.3. CA Intervention and Accountability	7
6. Distributed Ledger (Blockchain)	8
6.1. Chain Structure	8
6.2. Synchronization	9
6.3. Ledger Utility	9
7. Security Analysis & Justification	9
8. Limitations and Future Work	10
9. Conclusion	10

1. Introduction

The project develops a Peer-to-Peer (P2P) auction system that overcomes the dilemma of balancing anonymity and non-repudiation. The hybrid architecture eliminates the need for a central authority, using a distributed ledger to ensure consistency and fairness.

The system achieves privacy through RSA Blind Signatures, which allow bidders to authenticate themselves without revealing their identity. Simultaneously, it ensures accountability (non-repudiation) through an Encrypted Identity Escrow mechanism. This allows the winner to be identified and held accountable after the auction closes, while keeping all other bidders anonymous.

2. Requirements & Threat Model

Designing a secure P2P auction system requires balancing anonymity with accountability. This section outlines the functional constraints and security obligations derived from the project specifications, alongside the adversarial model governing our trust assumptions.

2.1. System Requirements

The system acts as a privacy-preserving peer-to-peer network where announcements and bids are broadcast to all users without a central auction logic server. Sellers publish auctions and bidders submit public bids while maintaining mutual anonymity until the winner is determined. To support this, the system guarantees:

- **Anonymity:** Identities remain confidential throughout the bidding phase.
- **Authenticity & Integrity:** Only registered participants can bid; messages are tamper-proof.
- **Non-repudiation:** Winning bidders cannot deny their bids.
- **Trusted Timestamping:** Verifiable timestamps resolve ties (earliest bid wins).
- **Selective Disclosure:** Identity revelation is strictly limited to the seller and winner.
- **Confidentiality:** All communications must protect content confidentiality and integrity.

2.2. Threat Model

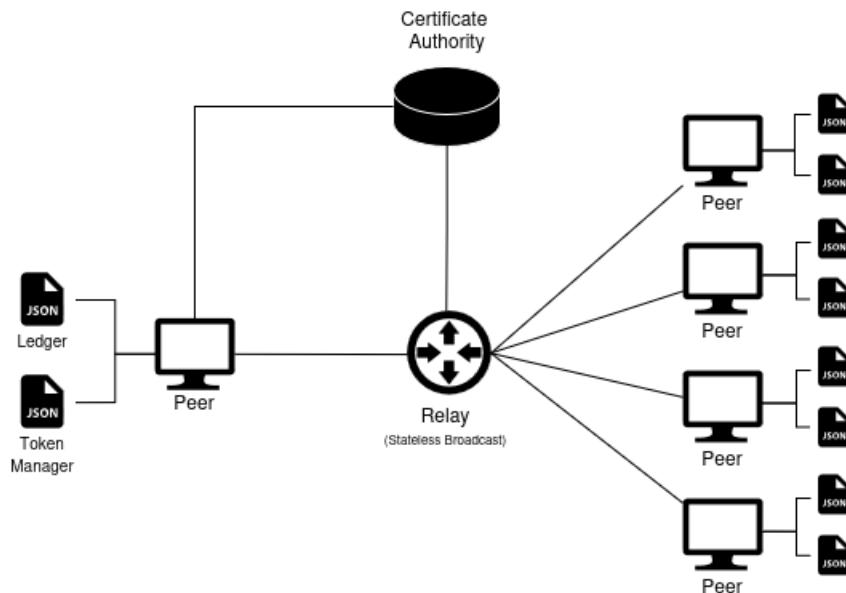
We assume a hostile environment where attackers can intercept or replay network traffic, necessitating end-to-end encryption. Malicious insiders (registered peers) may attempt token double-spending or bid denial, which is mitigated by the ledger and identity escrow mechanisms.

Regarding trust boundaries:

- **Relay Server:** Treated as an *untrusted* distributor. It observes metadata (IPs) but cannot modify content or influence validity.
- **Certificate Authority (CA):** Trusted for registration and timestamping. However, the use of *Blind Signatures* ensures the CA cannot link users to specific bids during the auction phase.

3. System Architecture

The solution designed for this auction system is inspired by Peer-to-Peer (P2P) network architectures. Although the initial objective was to achieve a pure P2P network, the system incorporates a Relay. The Relay (which may consist of multiple nodes) serves as a stateless broadcast node. This design choice enables full anonymity for participating peers. The star-like topology also integrates one trusted service: the Certificate Authority (CA).



Note: This architecture diagram is a simplified representation of the system. For clarity, some connections are omitted. All peers have a connection with the CA.

3.1. Components

The system architecture relies primarily on three main components: the Peers, the Relay, and the CA.

- **Peers**

The Peers are the core components of the system. They provide a command-line interface (CLI) for user interaction. Peers are responsible for sending messages into the network and serve as validators for the application logic. Each Peer maintains two main structures: a **Ledger** and a **Token Manager**.

Ledger: The Ledger is the local database for each Peer. This structure follows a Blockchain approach to ensure integrity. The Ledger is further explored in Section 6.

Token Manager: The Token Manager is the structure responsible for the creation and storage of tokens and their blinding factors. Token logic is detailed in Section 4.2 and Section 5.4.

- **Relay**

The Relay acts as the message broadcaster for the entire network. It is a stateless device that serves as the middleware through which all traffic flows. Upon receiving packets, the Relay broadcasts the message to all other Peers through TCP connections. Crucially, the Relay masks the Peers' IP addresses. Since traffic received by a Peer originates from the Relay, it becomes significantly harder to track source IPs, which are considered quasi-identifiers. Additionally, the Relay facilitates efficient broadcasting for messages sent by the CA. This component acts solely as a broadcasting device and it performs no application logic or validation.

- **Certificate Authority (CA)**

The Certificate Authority is the sole trusted service within the system. The CA operates as an HTTP server developed using the FastAPI library in Python. It provides a list of services crucial to the system's functionality.

Certificate Issuance and Peer Registration: The CA manages the admission of new users using the Certificate Signing Request (CSR) protocol. Upon receiving a CSR from a joining Peer, the CA verifies the credentials and transforms the request into a valid X.509 certificate, formally registering the Peer's identity within the system.

Blind Token Signing: The CA acts as the cryptographic signer for anonymity credentials. It signs blinded tokens submitted by Peers, enabling them to acquire valid and trusted tokens. This allows Peers to prove their legitimacy during auctions without the CA ever seeing the underlying token value.

Trusted Timestamping: Since distributed Peers may operate with non-synchronized local clocks, the CA serves as the authoritative time source. It provides verifiable timestamps for all critical events, ensuring a consistent chronological ordering of bids and resolving potential timing disputes.

4. Identity & Anonymity Framework

The Identity & Anonymity Framework defines who a user is within the system before they interact with the auction protocol. In contrast with traditional centralized systems, our design separates long-term identity from short-lived anonymous credentials, enabling strong privacy during bidding while preserving accountability when necessary. This dual model ensures that peers can authenticate themselves without exposing their real identity until the reveal phase.

4.1. Dual-Identity Model

Each participant in the system possesses distinct layers of identity that serve different security purposes:

Real Identity (PKI Based Certificate)

Upon registration, every peer generates an RSA key pair and submits a Certificate Signing Request (CSR) to the CA. The CA verifies the request, assigns a UUID, and issues an X.509 certificate binding:

- the peer's public key,
- the assigned UUID,
- the CA's digital signature.

This certificate constitutes the peer's long-term real identity and is used only during identity reveal or dispute resolution. Because certificates are never broadcast during bidding, they provide authentication without exposure.

Auction Identity (Blind Signed Tokens)

While the certificate establishes who the user is, it is not used directly in auctions. Instead, peers operate using anonymous auction identities, represented by blind-signed RSA tokens. These tokens authenticate a user's right to perform an action (auction creation, bidding, reveal), but they never reveal or leak the underlying certificate, ensuring that real identities remain unlinkable to their actions.

Group Identity (Shared Encryption)

All peers in the system share a Group Identity enforced by a common symmetric encryption key, which protects the confidentiality of the network against external entities. The CA distributes this key to peers upon registration. Additionally, this identity is managed strictly: when a peer leaves the network, the Relay notifies the CA, triggering a key rotation to ensure forward secrecy. The CA generates a new group key and distributes it to the remaining peers via a broadcast message containing a list of encrypted payloads: each payload consists of the new group key encrypted individually with the RSA public key of a specific active peer.

4.2. Blind-Signed Anonymous Tokens

Blind-signed tokens are the mechanism that allows users to participate anonymously while proving that they are legitimate and registered. The CA signs these tokens blindly, meaning it validates the request without learning the token's real value.

How users obtain a token ?

1. The peer generates a fresh random token_id.
2. This token is hashed (SHA-256) and transformed into an integer m.
3. A random blinding factor r is selected.
4. The peer computes the blinded value " $m' = m * r^e$ " using the CA's RSA public exponent.

5. The blinded token m' is sent to the CA.
6. The CA signs it " $s' = (m')^d$ " and returns s' without ever seeing m .
7. The peer unblinds the signature " $s = s' * r^{-1} \bmod n$ "
8. The tuple (token_id, signature, r) is stored and becomes the user's Auction Identity.

How tokens are used ?

The tokens are used in the sensitive actions, which includes **auction creation**, **bid submission**, **auction close events**, **winner reveal** messages. Peers verify token signatures locally, and the Ledger prevents duplicate token reuse, ensuring fairness and preventing Sybil attacks. Through this process, the user becomes authenticated but unidentifiable, wearing the "mask" that represents their anonymous auction identity.

4.3. Encrypted Identity Package (Revocable Anonymity)

Blind signatures alone provide anonymity, but the system must also guarantee accountability. To bridge this gap, every auction and bid message embeds an Encrypted Identity Package, a cryptographic escrow that ties a token to a real identity, without revealing it.

Each package includes:

- **real_uid** – the CA-assigned unique identifier,
- **cert_pem_b64** – the peer's X.509 certificate,
- **token_id_bound** – the token used for this specific action,
- **nonce** – a random value ensuring freshness.

The package is encrypted using a hybrid RSA–AES scheme:

1. The peer generates a random AES key and encrypts the package using AES-GCM.
2. The AES key is encrypted using the CA's public RSA key.
3. The final encrypted blob is attached to the message.

Only the CA can decrypt the full package, ensuring that identities remain protected unless the reveal process is activated. This mechanism ensures a balance between privacy and accountability, actions remain unlinkable unless opening is justified, provable, and cryptographically enforced.

5. Auction Protocol & Lifecycle

This section describes how users interact with the auction system once their cryptographic identities and anonymity mechanisms are in place. It details how auctions are created, how bids are formed and validated by all peers, and how the winner is securely determined and identified while preserving anonymity throughout the bidding phase.

5.1. Auction Creation

In the main menu, the user can create an auction using the input indicated along with the rest of the possible actions, which is **auction {name} {min_bid}**. Where *name* identifies the auctioned item and *min_bid* specifies the lowest acceptable bid.

Upon receiving these parameters, the peer constructs an auction message with the following fields:

- “**id**”: id of the action (generated procedurally)
- “**type**”: type of action (in this case, auction)
- “**name**”: name of the auction
- “**closing_date**”: date and time when the auction closes, bids are no longer accepted, and the winner is decided
- “**min_bid**”: minimum value that must be used in the bid to bid on the auction
- “**token**”: token data necessary for the action to be valid and taken into account
- “**public_key**”: public key of the auction

The auction message is encrypted with the group key and broadcast through the Relay to all participants. Each peer independently validates the token, timestamp and message structure before recording the event in its local Ledger.

5.2. Bid Message Format

Once another participant receives notification that an auction has been created, this user can create a bid with the command **bid {auction_id} {min_bid}**. Where *auction_id* is the ID of the auction for which they want to bid and *min_bid* the minimum amount the user must bid to have a chance of winning the auction.

Once these values requested by the program have been provided, a JSON message is created and sent to all other participants. This message has the following structure:

- “**id**”: bid ID
- “**type**”: type of action, in this case ‘bid’
- “**auction_id**”: auction ID
- “**bid**”: the amount of the bid
- “**token**”: token data required for the action to be valid and taken into account
- “**encrypted_identity**”: user identity encrypted with the CA public key

This message is encrypted using the group key and broadcast to all peers. Each peer validates it before appending it to the Ledger.

5.3. Validation Rules

Peers enforce strict validation rules for every auction and bid message:

- Any message shared with other participants, whether it is a bid, auction, or any other type, must have the data token.

- The data token used must be valid and cannot be reused in any other action.
- A bid is only accepted when:
 - The auction ID is valid (it exists and is not your own auction).
 - The `min_bid` is valid.
 - The bid is received within the time frame in which the auction is available. If a bid is received but the auction time frame has ended, the bid is rejected.

By enforcing validation locally, peers prevent a malicious Relay or participant from injecting or modifying state.

5.4. Winner Determination & Identity Reveal

5.4.1. Winner Determination & Identity Reveal

When the auction closes, the Auction Winner identifies the winning bid from their local ledger. To prove they are the rightful owner of the winning token, the winner sends a message to the Seller containing:

- The token identifier (`token_winner_bid_id`).
- The blinding factor (`blinding_factor_r`) used to generate the winning token.
- A session key (`deal_key`) encrypted with the auction public key $E_{apk}(deal_key)$.

The blinding factor and token ID are encrypted symmetrically using the `deal_key`. The Seller decrypts the `deal_key` with their auction private key and then decrypts the payload. The Seller then verifies the blind signature logic using the recovered blinding factor and token ID to authenticate the winner, ensuring non-impersonation.

5.4.2. Mutual Authentication (Seller's Reveal)

Once the winner is authenticated, the Seller performs a symmetric identity exchange.

- The Seller encrypts their own identity package (Certificate, etc.) using the same `deal_key` and sends it to the Winner $E_{dk}("Certificate", "blinding_factor_r", "token auction id")$.
- The Winner decrypts this package and verifies the certificate, achieving mutual authentication.

This process ensures that only the winner and the seller gain access to each other's real identities, preserving the selective disclosure property.

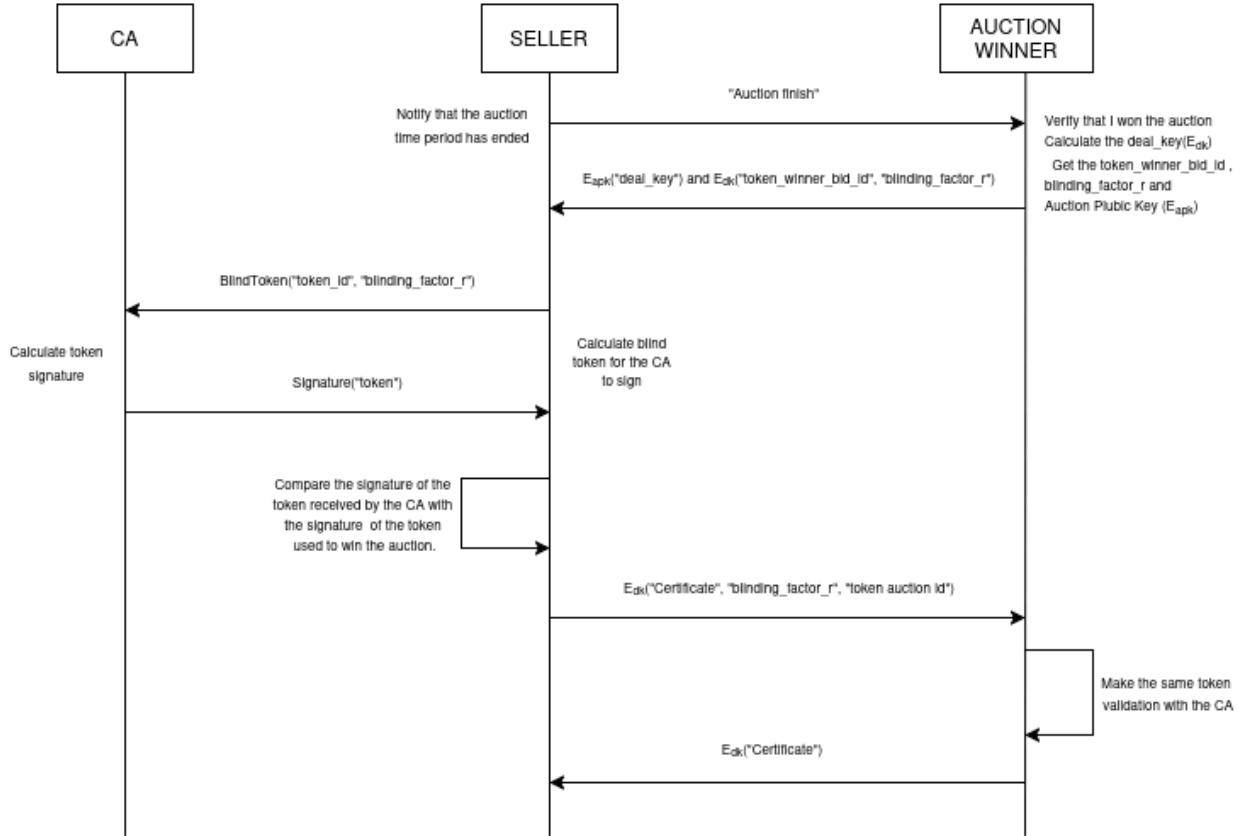
5.4.3. CA Intervention and Accountability

The CA acts as a final safeguard to ensure accountability and non-repudiation.

- If any party behaves dishonestly, refuses to reveal their identity, or if inconsistencies arise, the CA can intervene.

- The CA possesses the ability to decrypt all encrypted identity packages sent during the bidding process, which can happen in very strict conditions.
- By decrypting these packages, the CA can recover the real user identity behind a disputed token and issue a signed identity receipt, binding the identity to the action.

This balance guarantees that bidders remain anonymous during normal operation, but misbehavior can always be traced back to the responsible party.



6. Distributed Ledger (Blockchain)

The Ledger, as referenced in Section 3, serves as the local database for each Peer. By utilizing a blockchain-based data structure, it guarantees the integrity and immutability of the auction history across the distributed network.

6.1. Chain Structure

The Ledger is composed of a sequence of blocks linked by cryptographic hashes. To simplify the implementation for this project, each block contains a single event. This structure relies heavily on cryptographic hashing to guarantee integrity: each block includes the hash of the previous block (`prev_hash`) and its own self-calculated hash (`block_hash`). Additionally, each block contains its height in the chain, a timestamp, and the list of events it contains, one per block in this case.

6.2. Synchronization

When a new Peer joins the system, it requests the full Ledger from existing Peers. To ensure consistency in this decentralized environment, the Peer validates the hash chain of the received ledgers and adopts the longest valid chain (*Longest Chain Rule*). This ensures that new Peers synchronize with the correct global state and possess a valid history of all network events.

6.3. Ledger Utility

The Ledger acts as the authoritative source of truth for the system. Peers rely on the chain to perform critical state verifications, including:

- **Double-Spend Protection:** Peers query the Ledger to verify if a specific `token_id` has already been consumed in a prior transaction.
- **Highest Bid Verification:** The chain provides an immutable record of all bids, allowing Peers to determine the current highest bidder for any active auction.
- **Auction Ownership:** The Ledger stores the initial auction creation events, enabling Peers to link specific auctions to their creator's anonymous token.

7. Security Analysis & Justification

This section evaluates how the system satisfies the core security requirements defined in Chapter 2, mapping the architectural choices directly to the required security properties.

- **Anonymity** The system guarantees that seller and bidder identities remain confidential during the bidding phase. This is achieved by **RSA Blind Signatures**, which decouple the user's real identity from the tokens used in auctions. Additionally, the use of a **Stateless Relay** masks the IP addresses of the peers from one another, preventing network-level identification.
- **Authenticity & Integrity** Authenticity is guaranteed because only registered participants with valid CA-issued certificates can obtain the tokens required to interact with the system. Integrity is ensured at the network layer by **AES-GCM encryption** (which prevents tampering in transit) and at the application layer by the **Local Ledger**, where hash chaining makes any modification to past bids immediately evident.
- **Non-repudiation** The system guarantees that a winner cannot deny ownership of a bid. This is achieved through the **Encrypted Identity Package** (Revocable Anonymity) attached to every message. While the user remains anonymous to peers, this package allows the CA to cryptographically link an anonymous token back to a specific registered user if a dispute arises.
- **Trusted Timestamping** To resolve ties and ensure fair ordering, the system guarantees reliable timekeeping by using the **Certificate Authority (CA)** as a trusted

timestamping service. Peers rely on these signed timestamps rather than local clocks, preventing manipulation of auction closing times.

- **Selective Identity Disclosure** The system guarantees that identity revelation occurs only between the relevant parties. This is achieved via a **Hybrid Encryption** scheme during the reveal phase, which ensures that the winner’s identity proof is encrypted specifically for the seller’s private key, remaining inaccessible to all other participants.
- **Confidential Communication** The confidentiality of all client-server and peer-to-peer communication is guaranteed by the use of a shared **Group Key**. This symmetric encryption layer ensures that only registered, authorized system participants can read the broadcasted auction **data**.

8. Limitations and Future Work

While the current implementation successfully demonstrates a privacy-preserving P2P auction system, inherent limitations remain, most notably the centralization risk posed by the Certificate Authority as a “trusted opener” and the potential for race conditions due to the absence of a robust consensus algorithm. Furthermore, the system currently operates on a local network without network-layer anonymization, leaving IP addresses visible to the Relay. Future development should focus on decentralizing the infrastructure by deploying multiple Relays to improve fault tolerance, encapsulating application traffic within an anonymity network to mask physical identities, and implementing concurrency controls or a transaction pool to ensure consistent ledger synchronization under higher loads.

9. Conclusion

This project delivers a functional privacy-preserving P2P auction system that balances anonymity with accountability. By combining PKI-based registration, RSA Blind Signatures, and an Encrypted Identity Escrow, the system allows users to bid anonymously while ensuring that the seller and winner can securely identify each other at the end of the auction. A lightweight blockchain inspired Ledger provides integrity, ordering, and double-spend protection without relying on a central server.

The implemented design meets the key security requirements of anonymity, authenticity, integrity, non-repudiation and trusted timestamping. Although the system still depends on a trusted CA and lacks a full consensus algorithm, it forms a solid foundation for future decentralization and scalability improvements. Overall, the project successfully demonstrates how applied cryptography can support secure and fair auction mechanisms in adversarial P2P environments.