

A Cryptographically Secured Double Auction Framework for Perishable Agricultural Goods Trading

Mrs.Gini R

St.Xavier College of Engineering

Abstract—Because of the need for swift, equitable market exchange to reduce spoilage and minimize income losses for smallholder producers, perishable commodities (e.g., fruit and vegetables, livestock products and seafood) are very time-sensitive. The traditional wholesale channels of distribution limit producers to purchasers who are geographically close to them, allowing middlemen to exercise an inordinate amount of pricing leverage. This paper presents a blockchain-based, privacy-aware double auction system developed exclusively for the fresh produce sector. Instead of utilizing fixed posted prices, this system will conduct multiple bidding rounds in which buyers and sellers will be paired using an algorithm that takes market conditions into account, producing real-time results. The integrity and reliability of the auction are secured by a combination of four interlocking cryptographic mechanisms: (1) zero-knowledge succinct non-interactive proofs (zk-SNARKs) are used to verify the validity of bidder prices during the sealed bidding phase without revealing any information (in terms of monetary value) to any party; (2) linked ring signatures hide the identity of individual buyers during the open bidding phase, while providing complete auditability of amounts; (3) a hash-based commit-reveal protocol requires that each submission consists of two parts: (i) a binding commitment of the price, which provides confirmation of the bid; and (ii) a subsequent disclosure, making it mathematically impossible for a buyer to alter a bid after it has been submitted; (4) a behavioural scoring service continually monitors participant behaviour and dynamically updates trust ratings, discouraging non-cooperative or fraudulent behaviour over time.

Index Terms—agricultural e-commerce, blockchain ledger, commit-reveal protocol, double auction clearing, linked ring signature, zero-knowledge proof

I. INTRODUCTION

The worldwide trade of perishable agricultural products has a limited timeframe due to their biological nature rather than commercial factors. Produce begins to degrade only hours after

being harvested, which means that the amount of time a grower spends trying to find a buyer reduces both the quality and the revenue of the produce. Unfortunately, in many developing nations, the dominant method of exchanging fresh produce has been via traditional wet market or wholesale hub using opaque bilateral negotiation techniques to set prices rather than open competition. This gives a clear disadvantage to small growers who lack the bargaining power or market information necessary to challenge prices set by larger intermediaries. Some organizations estimate that approximately 25-33% of all produce harvested worldwide will suffer from preventable post-harvest loss, primarily attributed to delays in logistics and inefficiencies in marketing rather than inevitable biological processes. The widening of the pool of available counterparties through digital trading platforms will put a structural fix on these distortions while also automating various aspects of price discovery. However, the price discovery mechanisms that were observed in an investigation of available commercial agricultural e-commerce platforms were seen to have very rudimentary price discovery processes; for example; sellers will list their products at catalogue prices, buyers can choose to accept or reject the seller's price, and no method of establishing competitive prices is available through these mechanisms [4]. Research confirms that, through double auctions, which require both buyers and sellers to submit their respective bids and offers independently and to match those bids and offers with a clearing algorithm, we start to achieve price convergence toward competitive equilibrium, and thus achieve efficient price allocation and efficient welfare [5]. Thus, the real question is not whether double auctions can work, but whether or not they can be made to work in a geographically dispersed, technically untrustworthy, and distrustful counterparties' environment to be secure and private.

Web-based competitive auction systems have well documented weaknesses. The platform manager has access to the sealed bids; thus, they can control the clearing outcome in such a way that favours their selected counterparties. Visibility of identity on the side of buyers at auction opens the door to collusion among larger buyers. Furthermore, without a permanent audit trail, those participants who believe they have been treated unfairly have no means by which they can verify their claim [6]. Also, in the agricultural context where transactions where the individual size of each transaction may be relatively small, aggregate damage from periodic, systematic misconduct can seriously affect rural communities.

Distributed ledger technologies can help partially address these concerns through the creation and maintenance of a common, immutable record for all auction events that cannot be altered unilaterally by any one party [7]. However, simply having a transparent ledger does not assure confidentiality of bids since every transaction contained in a distributed ledger can be viewed by all parties with access to any of the nodes in the network. Therefore, to achieve the twin goals of auditability and confidentiality a layer of cryptographic methods is required that will enable the selective proof of certain characteristics of the data without revealing the actual data [8]. This paper makes the argument that zk-SNARKs, linked ring signatures and commitment schemes are precisely the cryptographic tools needed to enable the development of a cohesive and implementable auction platform. The primary contributions of this work are as follows:

- 1) A unified dual-mode auction engine accommodating both sealed-bid and open-bid rounds within a single web-based interface.
- 2) A zk-SNARK module that verifies bid validity without exposing monetary values to the auctioneer or other participants.
- 3) A linked ring signature scheme that conceals individual buyer identities during open-bid rounds while permitting linkability detection.
- 4) A hash-based commit-reveal protocol that eliminates the possibility of post-submission bid revision.
- 5) A behavioural reputation engine that scores trader conduct over time and applies graduated incentives to encourage honest participation.

The remainder of the paper is structured as follows. Section 2 situates the work within the existing literature. Section 3 describes the proposed architecture and each constituent algorithm.

Section 4 details the implementation choices. Section 5 evaluates performance and security properties. Section 6 draws conclusions and identifies avenues for future work.

II. RELATED WORK

2.1. Auction Mechanisms for Digital Markets

The theoretical foundations of double auction design were established and consolidated by McAfee [5], with their dominant-strategy mechanism creating conditions for the near-full efficiency of a bilateral exchange except for one excluded trade. The subsequent computational works have translated many of these same insights into practice, including Huang et al.'s [9] construction of a double auction for cloud resources, which cleared in polynomial time while satisfying the principles of individual rationality and budget balance, and Kang et al.'s

2.2. Distributed Ledger Applications in Agriculture

There has been a surge in academic research on using blockchain technology in agricultural and food systems. In this area, the work of Tian [11] is notable for creating an RFID-based provenance ledger that enables consumers to trace the entire journey of a product, from field to shelf; however, it does not address how prices are established in the supply chain. Aung and Chang [12] have a broader perspective on the need for transparency in food supply chains and propose that using smart contracts to trigger payments based on delivery confirmation can significantly reduce counterparty settlement risks associated with distance.

2.3. Cryptographic Tools for Auction Confidentiality

The cryptographic community has been looking at the issue of conducting both a secure and a privacy-protecting auction as an interesting challenge; Galal and Youssef [13], for example, implemented a Vickrey auction on the Ethereum blockchain with garbled circuits and were able to demonstrate that on-chain proof of auction results is possible, although expensive computationally. Groth's introduction of pairing-based succinct proof systems [14] has led to a

significant reduction in proof size as a result of having zk-SNARKs available for blockchain applications. The ring signature scheme developed by Rivest, Shamir and Tauman [15] provides group members with the ability to sign on behalf of their group without publicly identifying individual signers. Liu et al. proposed an extension [16] to ring signatures to make it possible to determine when an identity was used more than once while preserving the anonymity of all users. Also, commitment schemes have been utilized in order to prevent front-running (transaction re-ordering) in DeFi contexts [17]. Our contribution is bringing together all three of these primitives into a single, coherent agriculture-based auction platform and assessing the overall overhead rate incurred as a result.

III. PROPOSED ARCHITECTURE AND ALGORITHMS

3.1. Layered System Design

Three parts make up the auction platform's functionality: the presentation layer consists of the front-end UI delivered as a Bootstrap-responsive web app for presenting product auctions, demonstrating how to place bids, and displaying auction results requiring no platform-specific EXE files on the participant end; the application logic layer (Flask Route Handlers via Python 3.10) contains the logic necessary to process the state of the auction (transitions) and will also be used to call cryptographic sub-functions to enforce business requirements (reservation price floors and all bids received before an auction is concluded); the data persistence layer will be split into two (2) components — a MySQL RDBMS that will store off-chain records that can be indexed/queried (i.e., participant profile information, commodity metadata, and audit logs) and a hash-chained block ledger to provide non-repudiation/tamper-evident record of bid commitments, cryptographic proofs of winning bidders, and auction settled results.

3.2. Zero-Knowledge Proof Layer (zk-SNARKs) Sealed-bid participants must demonstrate that their offer b satisfies the auction's eligibility constraints

— specifically, that b lies within a declared range

$[b_lo, b_hi]$ — without disclosing b to any observer. This is accomplished by encoding the constraint as an arithmetic circuit C over a prime-order field and constructing a non-interactive proof under the Groth16 protocol. Letting pk denote the structured reference string for proving and vk the corresponding verification key, the prover computes:

$\pi \leftarrow \text{Prove}(pk, x, w)$

where the public statement $x = (H(b, r), b_lo, b_hi)$ encodes the bid commitment and range bounds, and the private witness $w = (b, r)$ contains the raw bid together with a random blinding factor r . The

verifier — executed by the auction engine at clearing time — checks $\text{Verify}(vk, x, \pi) \in \{0, 1\}$. The Groth16 instantiation yields constant-size proofs (two group elements on G_1 , one on G_2) and verification cost comprising three pairing evaluations regardless of circuit depth, making it well suited to resource-constrained deployment scenarios.

3.3. Identity Privacy via Linked Ring Signatures Procurers use an (open-bid) auction system when selling their goods. In an open-bid round of bidding, procurers can place bids for items they

want to buy. The procurers have a long-term key pair (x_i, y_i) , which was established during the onboarding process. A procurer constructs a signing ring, $L = \{y_1, y_2, \dots, y_n\}$, consisting of the public keys of all the currently active participants in the system. Using the signing ring, the procurer can create a linked ring signature (also referred to as a "linked-ring signature"), $\sigma = (c_1, s_1, \dots, s_n, I)$, where $I = x_i * H_p(y_i)$ is a deterministic key-image created from the private key of the procurer. Because any verifier can verify that σ was created by a member of the signing ring without learning the identity of the signer, the use of ring signatures provides anonymity for procurers. If there are two distinct linked signatures with the same key-image, then both signatures must have been created by the same procurer. Therefore, it is possible to detect whether or not a procurer has submitted a bid twice. The platform has enforced a minimum ring size of $n = 10$ to ensure that an appropriate level of anonymity is guaranteed.

3.4. Bid Integrity via Commit-Reveal Protocol

involves submitting your intended bid (b) with a randomly selected 'nonce' (r) as part of the commitment phase. When you submit your intention to bid you create a 'cryptographic digest' using a one-way function; this function is called SHA-256. You use SHA-256 to create a commitment digest from your intended bid and the nonce ($C = \text{SHA-256}(b + r)$). You then broadcast the digest (C) to the ledger before submission window closes. You are then able to use the values (b, r) to disclose your actual bid to the bidding platform. The platform will then use SHA-256 again to create another commitment ($\text{SHA-256}(b + r)$) and compare this value with the original commitment on the ledger.

3.5. Auction Clearing Procedure

At each round's conclusion, the he is executed using a modified McAfee-type of double auction clearing mechanism to serve the traders. The sellers' offers are sorted in ascending order of price, i.e., they form supply schedule S with elements $S =$

$\{(p^s_i, q^s_i)\}$, while buyers' bids are sorted in order of highest to lowest price, i.e., they form the buyers' demand schedule B with elements $B =$

$\{(p^b_j, q^b_j)\}$. The algorithm then can look at both schedules in parallel to determine the largest index k such that the k th buyer bid weakly beats the k th seller ask, (specifically $p^b_k \geq p^s_k$). If k is found then the uniform market clearing price P^* can now be calculated as: $P^* = (p^b_k + p^s_k) / 2$. Subsequently, all matched pairs of trades up to the k th index are executed at P^* , thus each of the matched individuals fulfil:

3.6. Behavioural Reputation Scoring

Where $\alpha = 0.85$. The event signal $\delta(t)$ reports the quality of the event outcome. In particular: - A completed trade results in $\delta(t)$ being given a value of 100; - Uncontested events that are deemed

neutral will each result in $\delta(t)$ receiving a value of 50; - Events associated with improper conduct (based on such conduct as failure to disclose within the commit/reveal window, the unilateral withdrawal from a bid or confirmed collusion) will each result in $\delta(t)$ receiving a value of 0.

IV. SYSTEM IMPLEMENTATION

4.1. Technology Stack

Our server codebase is written in Python 3.10 which has many great libraries helpful for cryptography and numerical calculations. All web transactions go through Flask's web server and are routed to handler functions depending on how a user has interacted with the system; whether they are updating their session status, invoking auction logic or retrieving JSON payload or HTML templates. All front-end user page content is created with bootstrap 5 plug-in components thereby providing a uniform mobile-friendly front-end without modifying the CSS for end-users. All persistent off-chain data e.g., user accounts, commodity catalogues, auction round meta data are stored in a MySQL 8 schema which is accessed via a parameterized query to prevent SQL injection attacks. Development/testing of the local environment for application development is performed using WAMP Server which contains Apache, MySQL & PHP as one Windows hosted

service manager. Two main libraries are utilized by cryptographic subroutine implementations. For hashing or hashing commitments and creating a link between blocks, the pycryptodome library provides SHA-256. The pycryptodome library also implements AES-256-GCM for encrypting user password credentials that are at rest. The elliptic curve mathematics behind both ring signatures and zk-SNARK proofs are provided by the ecdsa library. Numpy array primitives are used to perform the high-volume arithmetic required by the clearance and scoring algorithms. Panda DataFrames are used to batch process and export auction round data.

Table 1. Technology Stack Components and Assigned Roles

Component	Assigned Role
Python 3.10	Server-side logic; cryptographic subroutine host
Flask	HTTP routing; API endpoints; session management
MySQL 8	Relational storage for off-chain structured records
Bootstrap 5	Responsive, component-driven frontend rendering
WAMP Server	Integrated local deployment and service management
pycryptodome	SHA-256 hashing; AES-256-GCM encryption
ecdsa	Elliptic-curve ring signature and zk-SNARK operations
numpy / pandas	Vectorised clearing arithmetic; tabular data handling

Note: All listed packages are open-source and carry active community maintenance.

4.2. End-to-End Transaction Flow

A complete auction cycle passes through five ordered phases. During Enrolment and Key Provisioning, a new participant generates an ECDSA key pair on the secp256k1 curve; the public key is registered with the platform and published to the active participant list used for ring construction. In the Listing Phase, a grower publishes a commodity entry specifying produce category, available quantity (kg), quality grade, and a non-negotiable floor price. The Bid Submission Phase opens for a configurable window during which buyers execute either the sealed-bid workflow (commitment plus zk-SNARK) or the open-bid workflow (ring-signed bid message). At window close, the Clearing Phase invokes the McAfee procedure, computes the uniform price, and dispatches match notifications. Finally, the Settlement and Rating Phase records matched transaction identifiers on the ledger, triggers payment workflow hooks, and updates the reputation scores of all participants whose conduct was observable in the round.

4.3. Ledger Design

Rather than incurring the throughput limitations and token-cost overhead of a public blockchain, the platform maintains a purpose-built hash-linked ledger scoped to auction participants. Each block encapsulates four fields: a Merkle root aggregating all bid commitments submitted in the corresponding round; a set of (vk, π) pairs enabling post-hoc re-verification of every sealed bid; the complete clearing record specifying matched pairs, quantities, and the clearing price P^* ; and a signed vector of reputation delta values. The ledger is replicated in full to all registered participants, so any actor can independently verify the integrity of historical rounds by recomputing the hash chain from the genesis block. Tampering with any historical record invalidates the hash links of every subsequent block, making manipulation immediately detectable.

V. PERFORMANCE EVALUATION AND SECURITY ANALYSIS

5.1. Experimental Configuration

The system under test was a standard commodity workstation that had an Intel Core i7-12700 twelve-core processor running at a base clock speed of 2.1 GHz, 16 GB of DDR4 dual-channel memory and a 512 GB NVMe SSD with Ubuntu

22.04 LTS OS installed. The Flask application was deployed in production mode using Unicorn with 4 worker processes. Scripted clients were used to generate simulated auction loads, with concurrent participant pairs (buyers and sellers) ranging from 10 to 200.5.2 Cryptographic Primitive Overhead

The mean throughput and latency of the cryptographic functions tested over 1000 independent executions are shown in table 2. The latency of proof generation (Groth16) is the most time-consuming operation (average of about 1.84 seconds per operation), where the predominant cost is attributed to multi-scalar multiplication on the BN128 curve. This cost will be paid for by the user on the user's device each time they submit a bid, but there will be no additional cost to the server

that receives their bids due to this operation. The completion time for server-side proof verification is less than 40 milliseconds, which fits into the response time budget for asynchronous batched checks performed at the time of clearing. The amount of time required to create a ring signature grows linearly with the size of the ring; at the minimum required size of $n=10$ members, generating and verifying

.Table 2. Measured Latency and Throughput of Cryptographic Operations

Operation	Mean (ms)	Throughput
zk-SNARK Proof Generation	1840	0.54 ops/s
zk-SNARK Proof Verification	38	26.3 ops/s
Ring Sig. Generation ($n=10$)	12	83.3 ops/s
Ring Sig. Verification ($n=10$)	6	166.7 ops/s
Commit: SHA-256($b \parallel r$)	0.4	2500 ops/s
Reveal + Hash Re-check	0.6	1666 ops/s
Block Hash Computation	0.02	50000 ops/s

Note: Each figure is the arithmetic mean of 1000 trials. n denotes ring cardinality.

5.3 Clearing Latency and System Throughput

clearing an end-to-end transaction, including both opening submission windows and receiving settlement notifications once closed. We measured these total latencies across six different levels of participant counts. With 10 round-paired matches, the total round duration was 142 milliseconds; whereas with 200 participant-pair matches it took 987 milliseconds, indicating that total latencies increased in a manner consistent to that predicted using an $O(N \log N)$ function. Importantly, both examples were completed in less than one second— many financial industry practitioners use this duration as a minimum acceptable response time for internet-based trading applications.

Table 3. End-to-End Clearing Latency by Participant Count

Pairs (n)	Clearing (ms)	Bids/s	Ledger (ms)
10	142	140	220
25	218	229	235
50	341	293	241
100	589	339	258
150	784	382	263
200	987	406	271

Note: 'Ledger' = time to construct and append the round block to the hash-linked chain.

5.4 Adversarial Resistance Assessment

$C = \text{SHA-256}(b \parallel r)$; because the Groth16 proof gives only a guarantee of that the bid is a value within a range but does not provide any information about what the bid is, he will then have to conduct a pre-image attack against SHA-256 with an effective security level of 128-bit on the

hash. In the bid-manipulation scenario, an entity can not change their bid after viewing bids submitted by rival bidders without finding a collision in SHA-256, which is computationally impossible to do. In the identity-tracing scenario, a passive adversary passively observing the open bid stream will not be able to identify which ring member signed a particular bid; anonymity is unconditionally assured against any passive observer and is on a computational basis against any active observer, based on the Decisional Diffie-Hellman assumption. Double-bidding, submitting two bids from a single key, can be detected in $O(n)$ time by simply comparing key images.

5.5 Comparative Assessment

The proposed platform is compared to up to four existing reference systems using 6 operational capability dimensions in Table 4. The proposed platform is the only system found within agricultural trading contexts that combines competitive double auction price discovery with bid privacy

Table 4. Feature Comparison with Reference Systems

Capability	Proposed	AgriLedger	Tian [11]	Kang [10]
Double auction clearing	Yes	No	No	Yes
Bid-value confidentiality	Yes	No	No	No
Participant anonymity	Yes	No	No	Partial
Immutable audit ledger	Yes	Yes	Yes	No
Reputation mechanism	Yes	No	No	Yes
Agricultural specialization	Yes	Yes	Yes	No

Note: 'Partial' indicates pseudonymity without full cryptographic anonymity guarantees.

VI. CONCLUSIONS

The study describes the design and evaluation of a double auction system using cryptographic technology to meet the needs of fresh agricultural produce markets. It aimed to address the problem of smallholder producers (who are often economically disadvantaged compared to large producers) when negotiating with intermediaries without any transparency and how certain encryption technologies can support the building of an online marketplace that is verifiably equal, honest, and low-cost.

The prototype showed that the cost for participants to submit bn-for-range proofs is below 2 seconds, while servers verify these submissions in under 40 ms. The auction clearing process accomplished as many as 200 matched pairs of submissions in a total wall-clock time of 1 second, with ledger anchoring adding approximately 250 ms to the final time, regardless of the number of participants. The security analysis demonstrated that all four of the cryptographic mechanisms used in the prototype provided adequate protection against all five adversary scenarios relevant to online agricultural trade.

Future work may involve a number of extensions that can be built on top of this platform. For

example, deploying the ledger on a permissioned distributed ledger framework, like Hyperledger Fabric, would distribute trust across multiple node operators (e.g., agricultural cooperatives, rural banks, government regulatory bodies). This would strengthen the platform's independence from any one commercial operator. Another possible extension is to integrate the machine-learning demand forecast functionality into the auction configuration engine.

Conflict of Interest The authors declare no competing interests with respect to the research reported in this paper.

REFERENCES

- [1]FAO. The State of Food and Agriculture 2019. Rome: Food and Agriculture Organization of the United Nations, 2019.
- [2]Minot N, Sawyer B. Post-Harvest Losses and Gains from Reducing Them. IFPRI Discussion Paper 01488, 2016.
- [3]Gustavsson J, Cederberg C, Sonesson U, et al. Global Food Losses and Food Waste: Extent, Causes and Prevention. Rome: FAO, 2011.
- [4]Wang Y, Zhang H, Wang C. E-commerce adoption in agricultural product supply chains: A systematic review. *Journal of Rural Studies*, 2020, 75: 1–14. DOI: 10.1016/j.jrurstud.2020.01.003.
- [5]McAfee R P. A dominant strategy double auction. *Journal of Economic Theory*, 1992, 56(2): 434–450. DOI: 10.1016/0022-0531(92)90091-U.
- [6]Nisan N, Roughgarden T, Tardos E, Vazirani V V (eds.). *Algorithmic Game Theory*. Cambridge: Cambridge University Press, 2007.
- [7]Nakamoto S. Bitcoin: A peer-to-peer electronic cash system. Whitepaper, 2008.
- [8]<https://bitcoin.org/bitcoin.pdf>.
- [9]Goldreich O. *Foundations of Cryptography, Vol. 2: Basic Applications*. Cambridge: Cambridge University Press, 2004.
- [10]Huang J, Shin K G, Yu Y. RADAC: Rapid and accurate double auction clearing for cloud computing resources. In *Proc. IEEE INFOCOM*, Toronto, Apr. 2014, pp. 1–9. DOI: 10.1109/INFOCOM.2014.6847986.
- [11]Kang J, Yu R, Huang X, et al. Incentive design for reliable federated learning via reputation and contract theory. *IEEE Internet of Things Journal*, 2019, 6(6): 10700–10714. DOI: 10.1109/JIOT.2019.2940820.
- [12]Tian F. An agri-food supply chain traceability system combining RFID and blockchain. In *Proc. 13th Int. Conf. Service Systems and Service Management (ICSSSM)*, Kunming, Jun. 2016, pp. 1–6. DOI: 10.1109/ICSSSM.2016.7538424.
- [13]Aung M M, Chang Y S. Traceability in a food supply chain: Safety and quality perspectives. *Food Control*, 2014, 39: 172–184. DOI: 10.1016/j.foodcont.2013.11.007.
- [14]Galal H S, Youssef A M. Trustee: Full privacy-preserving Vickrey auction on Ethereum. In

Proc. Financial Cryptography and Data Security Workshops, Curaçao, Feb. 2018, pp. 214–223.

- [15] Groth J. On the size of pairing-based non-interactive arguments. In Lecture Notes in Computer Science 9666, Springer, Berlin, 2016, pp. 305–326. DOI: 10.1007/978-3-662-49896-5_11.
- [16] Rivest R L, Shamir A, Tauman Y. How to leak a secret. In Lecture Notes in Computer Science 2248, Springer, Berlin, 2001, pp. 552–565. DOI: 10.1007/3-540-45682-1_32.
- [17] Liu J K, Wei V K, Wong D S. Linkable spontaneous anonymous group signature for ad hoc groups. In Lecture Notes in Computer Science 3108, Springer, Berlin, 2004, pp. 325–335.
- [18] Bigi G, Miller A, Shi E, et al. Formal verification of smart contract security properties. In Proc. Financial Cryptography and Data Security, Christ Church, Feb. 2018, pp. 523–540.