**Ambiguity Resistant Privacy-Preserving Auction (ARPAN)**

 [dubiety](https://www.thesaurus.com/browse/dubiety)

**Best green**

**Red to be checked**

**Orange needs rewriting**

**Abstract**

Auctions have become very popular for trading a good, primarily when the actual value of that particular good is unknown in the market. With the internet era, auctions have become more fluent and easily conductible online; therefore, many online platforms have come into the picture. However, the primary concern with these platforms was that information such as bid values shared by the bidders with these platforms could be used to manipulate the results of the current or future auctions as well data privacy concerns are also there. To ensure safety from these issues, Privacy-preserving actions have been introduced such that bidders’ privacy is preserved. However, the possibility of bidders’ collusion to make the auction result ambiguous and auction results manipulation has not been considered well in privacy-preserving auctions.

Since privacy-preserving auctions hide most of the information to preserve privacy and there may be ambiguity in the auction result, the verifiability of auction results is another significant concern. In this paper, an ambiguity-resistant privacy-preserving auction (ARPAN) is proposed which ensures safety from bidders’ collusion together with the verifiability of the auction results and all other necessary properties. ARPAN is based on Secure Multi-Party Computation and Homomorphic Encryption, which enables us to perform an auction without disclosing the bid information of buyers to the auctioneer ensuring that bidders‘ collusion to change the auction result will be identified as well as the auction results will be verifiable to each participant. Thus, ARPAN preserves the bidders’ privacy, removes ambiguity in the auction result, and makes the auction result publicly verifiable.

**Introduction**

Auctions have emerged as one of the most excellent tools to gather the actual value of any product. For this purpose, various auction methods have been proposed in the past [ reference, precise not book] [2]. Popular auction types, such as English and Dutch [Reference], are frequently used. Most auctions prefer open bidding, i.e., bid value is open to each participant of the auction, and everybody knows what value a specific bidder is willing to pay for the auctioned good. The drawback with such auctions is that the bidders’ data is easily accessible to everyone, ultimately leading to unfair auctions having bid repudiation, false valuation, fake bidding, etc. [reference]. The second price auction [reference], where the highest bidder pays the second highest price, was proposed to solve many such issues, and the concept of sealed bidding was introduced. Due to the sealed bid, only the auctioneer can open and compare all the bids and then declare the winner. Sealed bidding removes the issue of price changes and the temptation to raise the price by limiting the openness of bid values to the auctioneer only. Still, the problem is that we must consider the auctioneer as a trusted entity and trust the auctioneer not to misuse bid information.

Modern internet-based auctions, i.e., e-auctions, platforms, [reference] are generally privately owned and led by the auctioneer itself, thus creating great opportunities to gain profit from using bidder’s data, for example, maliciously using it for upcoming auctions, in case of second priced sealed bid auctions auctioneer may change the second highest bid and make it too close to the highest bid and earn the difference as his profit, and it will be undetectable because the bids are sealed. In case the bids are not hampered by the auctioneer and certain bidder wins the auction and pays the second price as a winning price, another threat that is formed is on the subsequent auction of the same good; the auctioneer sets the reserve price of the good as the last auction's winning price thus securing his personal gain. The auctioneer may sell this data to third-party retailers, and they may use this information while selling the same good to the bidders [reference of data selling paper]. They may not sell goods lower than that bid price. Another major threat in a privacy-preserving auction environment is to ensure anti-collusion, two or more bidders may collude with each other and try to change the auction results or may try to know the bid values of other honest bidders and in most of the Plaids are in encrypted form and detecting such maliciousness by bidders is not easily detectable. The cases of auctioneer performing malicious behavior is not only on papers there are real life proof in the domain of energy trading where auctioneer misused the bidder data maliciously for the financial incentives [ reference 7 aur 16 from drive paper].

Mentioning above issues in case of privacy preserving auctions below listed properties must be ensured:

* Privacy: Privacy refers to preserving the bid values from the auctioneer and all other participants before and after the auction.
* Accountability: This means a malicious bidder can't disrupt the integrity of the auction results by submitting a deceitful bid value and can bypass the system's safety.
* Integrity: The Bidders can submit the bids within an allocated time frame, thus preventing the false submission after the time frame.
* Non-Repudiation: Bidders cannot deny the bid that they have submitted to the system; protocol should be able to map the bids to original bidders.
* Fairness: The bidder has the same power and authority as other bidders to check the authenticity of the results.
* Verifiability: Each bidder must be able to verify the auction results securely.
* Transparency: The bidders must guarantee the winning bid's origin, authenticity, and legitimacy.
* Anti-Collusion: Two or more bidders must not be able to collude with each other and try to disrupt the auction results or maliciously know the bid value of other bidders.

In this paper we present ARPAN which is PPA satisfies all above mentioned properties. The rest of the paper is organized as follows. Related works have been concluded in section 2. Section 3 provides the background of tools used to fulfill our work. The work of the proposed scheme is shown in section 4. Section 5 consists of the experimental evaluation and mathematical proof of the algorithm's security. Section 6 concludes the work. Next, we present a case study of various threat models against the proposed PPA.

**Literature Review**

PPA have been greatly considered in literature with emphasis on issuing public verifiability of the results and avoid auctioneer’s maliciousness, various trust models have been considered to do so. Table 1 presents the comparison of different models on the basis of trust model used and necessary auction properties satisfied by them.

In one of the early works [[ref](https://people.eecs.berkeley.edu/~tygar/papers/Multi-round_anonymous_auction_protocols/IEIC_multiround_anonymous_auction_protocols.pdf)], a multiple-round-based auction protocol was proposed with the concept of multiple () auctioneers, making the system safe with the threshold of () such that these number of auctioneers are not colluding with each other till than the system is safe. In the initial phase prices are shared for the auction goods and the bidders generate bid vectors for these prices. For the interested price they put their and for non-interested price they put 0, together with the random padding generated through sum of random polynomials. These bid vectors are than input to a SMPC algorithm. For the wining price number of present at the winning price is revealed and this is done by auctioneer using La-Grange scheme to solve the simultaneous equations and obtains the free variable, which gives the sum of identities of bidders who are willing to bid at that price. If a single bidder was interested for the highest price than only single is present and if sum of multiple is found than multiple bidders are interested at that price which is a case of tie and again subsequent rounds are performed with more refined price list near to highest price to get a single winner at wining price. If value of is very small than many subsequent rounds of auction will be required increasing the computation cost and dependency on threshold of auctioneers also increases communication cost. Verifiability of the auction results by each participant hasn’t been ensured in this work. Further this work was improved in [[ref](https://ieeexplore.ieee.org/document/884610)] by using the masking of bids in random polynomials which are generated for each bid by the set of distributed servers ensures trust in the system. The winner detected in same way as earlier but this time the are masked by the random polynomials. When the auctioneer declares a wining price all the distributed servers collaborate and remove the noise from these IDs to know the exact winner ID no other information is revealed to the auctioneer. This work improved the shortcoming of earlier work in terms of privacy, security, and second highest bid is kept secret from the winner in comparison to the previous work through the masking step by the servers. But in this work also auctioneer can know the highest wining price and verifiability by each auction participant is not ensured. The threat model has been shifted to servers from auctioneer in comparison of previous work. Instead of auctioneer threshold here server threshold has been used such that n-1 servers should be honest to complete auction safely. Another work that was presented in succession by same authors is given in[[ref](https://sci-hub.se/https:/link.springer.com/chapter/10.1007/3-540-46088-8_27)] a M+1 auction scheme is proposed where result verifiability has been assured by the side of the auctioneers which can be considered as partial verifiability, also it does not prevent the anonymity of the winning bidder from the auctioneers, and it requires larger number of auctioneers to fulfill the auction as mentioned in [[ref](https://sci-hub.se/10.1007/3-540-45664-3_8)]. #SAMEAuthors in [[ref](https://sci-hub.se/10.1007/3-540-45664-3_8)] use ElGamal encrypted bidder-generated bidding vectors consisting of bids encrypted by a public key which is handled by a set of distributed authorities in a threshold manner. For each bid bidders submit differential of their bid values as proof of their casted bid. auctioneer publicly computes the integrals of these differential of bids submitted by the bidders to verify the bids. Mix and match. Later auctioneer declares the winner. In this protocol the highest bid is reveled to the auctioneer after completion of the auction.The work in [12] is similar work that we are proposing is considered. It proposes a first price auction protocol where the concept of distributing the bid value to multiple shares is used thus distributing the trust to the bidders itself is considered. In this work role of auctioneer is played by the seller, so no auctioneer is involved during the whole process of auction thus it is shifting the threat model from auctioneer to seller, this paper explored bidder’s collusion possibility and a threshold of bidders being dishonest is considered to avoid bidders’ collusion and resulting in colluding bidders knowing the bid value of other bidders. Another threat is if the seller becomes malicious and colludes with the bidders than he has to collude with at least bidders to know other bids. This work also lacks in giving the auction results public verifiability to know whether or not bidders’ collusion or any corruption in protocol happened in between. If this work is applied in second price environment than it will be impossible to ensure that the seller changed the second highest bid for his profit as this work doesn’t ensures verifiability of the auction results.

In the works of [[ref](https://link.springer.com/chapter/10.1007/978-3-642-03549-4_20)] use multi-party computation based on secret sharing to develop a practical double auction. Their scheme uses verifiable secret sharing involving representatives of buyers, sellers, and the research project itself. Traders submit bids and asks rep- resenting how much they are willing to buy or sell at all possible prices. The bids and asks are then secret shared among the three servers for aggregation. Each server verifies that their received share is correct by the verification property of verifiable secret sharing. The servers then aggregate the individual shares to construct demand and supply curve shares. The parties compute the market- clearing price using secure comparisons on secret shared values. After traders submit their offers, no interactivity is required (their representatives interact on their behalf), and traders can submit multiple offers. However, the protocol does not allow traders to verify the results independently, and corrupting two out of three parties renders the protocol insecure.

In the works [[ref](https://dl.acm.org/doi/abs/10.1145/3600160.3600190)] a double auction is proposed based on the homomorphic encryption and zero knowledge proof of consistencies and it satisfies major auction properties such as pseudonymity, unforgeability, traceability, and non-repudiation. The scheme is fulfilled by the assumption of non-colluding third agent

Privacy preserving auction have found great importance in the domain of auction applications such as spectrum allocation, energy trading, data trading etc. One such work of [[ref]](https://ieeexplore.ieee.org/document/8384016) an application of privacy preserving auction has been proposed through which trading of big data is completed. In suggested protocol an intermediate platform is consider and a single auctioneer is considered with an assumption of both parties being independent of each other. Bidders bid their bidding price and forward it to the intermediate platform by encrypting with auctioneer public key. Intermediate platform further adds a padding of common random number homomorphic to all gained bids and transfers these padded bids to the auctioneer and on the basis of these padded bids winner is declared. Even though this model fulfilled the auction but the assumption of independence between auctioneer and intermediate platform doesn’t hold in real life, it they both collude than auctioneer can easily gather all bids in original form also public verifiability of the auction results is not ensured.

Another recent work in domain of energy trading is [[ref](https://www.sciencedirect.com/science/article/abs/pii/S0306261923000284)] where privacy preserving model has been used to fulfill energy trading without disclosing any confidential information to the auctioneer about bidders. This proposed architecture is based on blockchain and it requires a certifier who certifies the participant, creates unique ID, s and paillier homomorphic key pairs for the bidders and the auctioneer. Further each bidder prepares a padded information containing its original bid and a random number in form of product, further this information is encrypted with auctioneer’s public key. This information is not directly sent to the auctioneer rather it is propagated to each bidder present in the auction who further multiplier his random number to this information homomorphically and then forward to the auctioneer. Auctioneer collects all such padded bids and compares these padded bids with the help of secure two-party comparison protocol (explained in section 3) and declares the winner. This model of PPA lacks the public verifiability and fails when two bidders collude with each other. Even a single bidder can make the auction results incorrect and it will not be detectable. We have used this PPA model in our paper as base and further improved it in terms of public verifiability, bid privacy, anti-collusion etc.

**Preliminaries**

The section briefly introduces the tools considered in the proposed privacy-preserving auction scheme.

**3.1 RSA(Rivest–Shamir–Adleman) cryptosystems**

RSA [reference] is a type of asymmetric encryption that is based on the idea of the computational complexity of factorizing the product of large random prime numbers. it consists of two keys one public and the other private. The public key is used to encrypt the data and the private key is used to recover that data. One who holds the private key can decrypt any message encrypted with the private key of the same. Digital signature is a very famous example of application bass on such cryptosystems.

**3.1 Digital Signature**

A digital signature[reference] is a type of mathematical scheme where a recipient of the message can be sure that a particular message comes from a genuine source. For this purpose, a sender encrypts the message with the private key and the receiver can verify the legitimacy of the message by decrypting that message with the sender's public key. Let's say Ram sends a message to Shyam to know the genuineness of the message ram attaches a digital signature with the original message by encrypting the message with its private key and Shyam decrypts this encrypted text with Ram’s public key if the message and decrypted message are same than Shyam can be assured that message is authentic and comes from Shyam.

**3.1 Paillier Homomorphic Encryption**

It is a probabilistic asymmetric algorithm on public Key cryptography. The mathematical operation that can be performed over a ciphertext without decrypting it is the main advantage of Paillier homomorphic encryption [15]. Encryption is secure and based on a nth residue problem, making it difficult to compute the plaintexts. The main functions that are available in this scheme of cryptography are KeyGeneration(), Encryption(), and Decryption().

KeyGeneration(): This function creates two pairs of keys: a public key (pubKey) and a private key(priKey). To create a pair of keys, two large random prime numbers (p, q) are selected so that their greatest common divisor, i.e., gcd(n, (p-1)(q-1)), is one where n is pq. We find the lowest common multiple = lccat

m(p-1, q-1). Now we select a random number g such that it belongs to [2, n2] we do so to ensure that n divides the order of g by checking the expression,μ = (L(g l mod n2)-1) mod n where L is the function such that L(x) = (x-1)/n.

Encryption()/E(): For encryption of any text, we use the public key, which is(n, g). The plain text will be represented with m, which belongs to [0,n). We select a random number r that belongs in (0, n) to encrypt the plain text, and gcd (r, n) is 1. ciphertext c is computed as c = gmrn mod n2.

Decryption()/D(): For decryption, we use the private key, which is (l, μ). For the ciphertext c to be decrypted and computed the plaintext m, we do it with the help of the following expression m = L (cl mod n2) μ mod n.

The following equations, (1), (2), and (3), are the specific ways to perform algebraic operations over the ciphertext generated through this cryptosystem. The results are the same on decryption as if we have done these operations on the plain text directly.

D(E(m1)|pubKey E(m2)|pubKey mod n2)|priKey = (m1 + m2)mod n (1)

D(E(m1)|pubKey gm2mod n2)|priKey = (m1 + m2)mod n (2)

D((E(m)|pubKey)kmod n2)|priKey = (km)mod n (3)

When we multiply two ciphertexts and then decrypt them, the result is the sum of the two original plain texts shown in equation (1). The second part, i.e., equation (2), shows that if some number m2 is raised to the power of g and then multiplied to the ciphertext, then on decryption, the result is again the sum of the two numbers. The last equation, i.e., equation (3), shows that if some number k is raised as the power of the ciphertext, the result is the product of both plain texts.

**3.2 Secure Multi-Party Computation**

SMPC [reference] is an efficient way to know whose bid is the highest without disclosing the actual value of the bid. SMPC will help sort the ciphertexts of the bid values, and based on the result of sorting, the auctioneer will decide who is the winner.

**Secure Two-Party Comparison (STPC)**

In our proposed protocol, we have used a secure two-party comparison protocol to compare the bid values of two bidders. This protocol is performed between every bidder. Let's say bidder A and bidder b generate bid values b1 and b2. bidder A and B don’t want either the auctioneer or other bidders to know their bid value, and for this purpose, a mathematical formulation is performed, where both bidders generate a random value X collaboratively, and then it is padded to both b1 and b2, thus generating the expressions Xb1 and Xb2. Here, the auctioneer can easily compare both expressions, and no secret information is revealed to him or other bidders. The same mathematical formulation will be performed for each bidder, thus preserving the privacy of the bid and yet making it possible to compare the bid values.

The notions that we have used throughout the paper have been described in Table 1.

|  |  |
| --- | --- |
| **Symbol** | **Meaning** |
|  | Id of bidder |
|  | Bidder i |
|  | The random number of the bidder |
|  | The bid value of the bidder |
|  | The public key of the auctioneer |
|  | The private key of the auctioneer |
|  | Public key of bidder |
|  | The private key of bidder |
|  | Message is homomorphically encrypted with the public key of the auctioneer |
|  | Message is homomorphically encrypted with the public key of bidder |
|  | The decryption of the ciphertext by the public key of bidder |
|  | Message encrypted with RSA public key of bidder |
|  | The decryption of ciphertext with the RSA private key of bidder |

**4 The Proposed Privacy-Preserving Auction**

The section introduces the entities involved in the proposed auction and then discusses the proposed model by discussing various phases in the proposed auction scheme. The whole auction concludes in a phase-by-phase manner and each phase has its timeout before which each bidder has to ensure the completion of the phase on its side. Those unable to complete in the given time bound will be discarded from the auction.

**4.1 Entities involved in the proposed auction**

The privacy-preserving mechanism that we have proposed contains a total of four entities that are described below

* Auctioneer: The auctioneer is a single party that provides the platform for conducting the auction. The auctioneer is also an active participant and calculates the auction result based on the received ciphertexts through the bidding phases.
* Bidders: Bidders are the interested buyers of the seller’s product. In our setting, they are considered dishonest agents who may not follow the auction regulations and may try to collude with each other.
* Seller: The seller is the entity here to sell an item to interested buyers and is considered an honest agent. The seller’s interest is only in selling the product at the best price that the seller can get from the auction.
* Certifier: It is an entity that has been considered to generate the unique IDs for Auction and pallier homomorphic key-value pairs for all the parties involved in the auction. This entity is trustworthy to all the parties involved.

Makor concern of previous PPAs was the dependency on the central authority to avoid dependency on a central authority we have introduced the certifier and the data that is shared with the certifier is only the unique IDs of the participants to generate homomorphic key pairs. Here, the auctioneer can be any online auction platform service provider responsible for conducting the auction. We aim to achieve an auction where no sensitive information can be disclosed, thus preventing bid privacy for the auctioneer and externally interested parties, nor can bidders submit false data during SMPC to win the auction or disrupt the auction’s legitimacy thus making the auction collusion-proof. In our model, we consider both the bidders and the auctioneer dishonest. They are both interested in their benefit; bidders can try to submit false data and win the auction or to know the bid value submitted by other bidders, and the auctioneer wants to know the original value of the bids to earn benefit from this data in the future. The seller is considered an honest party only interested in selling the goods at the best price. The certifier is considered to make the scheme work, but the information shared with the certifier is kept to a minimum.

**4.2 High-level overview of the phases**

The protocol consists of a total of 5 phases, as depicted in Figure 1. Following is the high-level overview of the proposed protocol

1. Registration Phase: Each participant registers with the certifier and ensures their entry to the database. The certifier issues the public and private keys to each bidder and the auctioneer.
2. Bid Submission Phase: Bidders and auctioneers generate a large secret random number for themselves, which will be used to fulfill the STPC. Each bidder encrypts its bid value padded with its random number by the auctioneer's public key, together with the digital signature, and then forwards this data to each bidder by encrypting with their public keys. Each bidder multiplies his random number to the received information homomorphically and then forwards the same to the auctioneer for STPC
3. Bid Reshuffling Phase: The Auctioneer collects and creates a copy of all the received messages at its end and then multiplies its random number to each received message and propagates this message to the bidder to multiply their random number on this message.
4. Ambiguity Removal Phase: In this phase, ambiguity removal takes place; the auctioneer analyzes all collected data from the bid reshuffling phase to check if no collusion or wrong entry has been made through the bidders
5. Market Clearance This phase is for the declaration of the winner by comparing the bids generated via STPC and then submitted in the Bid submission phase later auctioneer makes the auction data public so that each participant can verify the auction results.

The in-depth discussion will of each phase will be made in the following subsections. In our approach, we present a secure two-party evaluation-based PPA where the possibility of bidders' collusion can easily be detected and a proper method to verify the auction results. Papers such as [references] have used the same approach as ours but lack anti-collusion and verifiability to discuss some paper that lacks some properties in comparison to our paper. The goal is to conduct an auction without disclosing any of the bid to the auctioneer, or the certifier.

**4.3 Phase 1: Registration Phase**

This phase is the preliminary phase, where an auctioneer submits the auction issue request to the certifier and submits the auction details to the issuer. Details can be time frames of the auctions, rules, protocols of the auction, date, and goods to be auctioned. The certifier issues a set of public and private keys of pallier homomorphic encryption and RSA encryption to the auctioneer and all the bidders willing to participate on the auction date. Here, the certifier works as the bulletin board, and no confidential information is shared with the certifier except a unique ID to generate the keys.

**4.4 Phase 2: Bid Submission Phase**

Bidders submit their bid as the sum of their own generated random number and their actual bid. Further, this data with digital signature is submitted to other bidders as represented in equation 4.

(4)

Here, the first part of equation 4 is the bid of bidder padded with its random number , and this information has double encryption, one with auctioneer public key and then further with bidder to whom this information is sent for STPC i.e., bidder public key . The Second part is the digital signature of on the sent information to maintain the authenticity of the sent information.

In the next step decrypts the information and adds its random number to the ciphertext homomorphically; together with this, bidder also adds its digital signature and submits this data to the auctioneer. The data submitted to the auctioneer by bidder that was gained to him from bidder is shown in equation 5.

(5)

The information from equation 5 is verified by the auctioneer with the help of digital signatures and then saved in the auctioneer’s database.

Algorithm 1 is the demonstration of the protocol that is followed during the bid submission phase.

**Algorithm 1**

1. For a set of bidders ,each bidder generates padded bids information at own end and shares with, each bidder .
2. Bidder decrypts and adds its random number to received at their node and then updates the information which is represented by

1. Auctioneer decrypts received and stores it to use in upcoming phases.

Bid submission prevents non-repudiation and privacy of the bids. Digital signature prevents any hampering to the data ensuring non-repudiation while the padding of random numbers ensures that no privacy of bids is leaked to the auctioneer when the bids get finally submitted to the auctioneer.

**4.5 Phase 3: Bid Reshuffling Phase**

As in the bid submission phase bids privacy is preserved and non-repudiation is prevented and the auctioneer can declare the winner based on secure two-party comparison protocol by using and papers such as [reference applied energy] have used the above same approach to declare the winner but this protocol doesn’t ensures anti-collusion if two bidders collude they can easily make the results of the auction incorrect or may gather the information of bid submitted by other bidders and that’s why phase 2 solely doesn’t guarantees that no such maliciousness has occurred in between, and neither provides any way to detect such maliciousness. Phase 3 proposed in this paper is to ensure the system safety from the above-described issue of bidder’s collusion and a method to verify the auction results. This protocol works on reshuffling the bid and taking the bidder's random number once again on their own generated message along with the random number of all other bidders without disclosing the message identity to them. The protocol works as depicted in algorithm 2.

**Algorithm 2**

1. Auctioneer updates all by adding its random number to it thus generating .

= +

1. Auctioneer forwards this along with the of all other bidders other than bidder k to bidder k
2. decrypts the first part of the information and add its random number homomorphically and then encrypts it with the next given public key.

1. Same is followed by each bidder and ion last the along with the sum of random numbers of all other bidders is received to the auctioneer as the last is of the auctioneer.
2. New gathered data by auctioneer after addition of all bidders’ random numbers is represented by
3. The auctioneer compares through the secure two-party comparison for each pair of bidders and decides who is the winner and keeps it secret with him till the market clearance phase.

This phase ensures that each bidder adds its random number to the padded bids submitted by each bidder and during this whole process the identity of submitted bid during the bid submission phase is not revealed which will be beneficial in upcoming the phase. If the identity of submitted bid is revealed than the bidders who have colluded in first phase will identity the message and again submit the same false data and we have to avoid it from being happening that is why we have maintained the anonymity of the submitted bid during this whole process.

**4.6 Phase 4: Ambiguity Removal Phase**

In this phase before the declaration of the winner auctioneer checks for anti-collusion due to the bidders. The auctioneer follows algorithm 3 to find the bidder's collusion that can took place in between. If the auctioneer finds any collusion than he rejects the auction and if no collusion happened than declares the winner, which is described in market clearance phase. Following is the description of algorithm 3.

**Algorithm 3**

1. Auctioneer collects all and checks the following equation for every iteration of
2. N

Where the value of

1. Auctioneer checks thevalue on the basis of following conditions

N

This phase removes any ambiguity happened in between the earlier phases this phase bounds the bidders to submit true input during the bid submission and bid reshuffling phase otherwise it gets detected in ambiguity removal phase using algorithm 3. Earlier works which used these approaches lacked detection of such ambiguity and if such ambiguity goes unnoticed than it is guaranteed that the auction results will not be correct. Two colluding bidders can easily take the auction offtrack of the auction rules. They can also extract bid values of other bidders. But by the use of the algorithm 3 the auctioneer can find any such ambiguity present in the collected bid data.

We have also provided the proof of above algorithm that it will work even if a single bidder will create ambiguity in between the bid submission. The proof has been presented in the case study section.

**4.7 Phase 5: Market Clearence Phase**

After the ambiguity removal phase auctioneer sorts all collected and finds the winning bidders. For declaration of the winning bidder the auctioneer encrypts the winning bidder’s with the winning bidders RSA public key and propagates it to the network. Each bidder tries to decrypts this message by the auctioneer and only the winner bidder can decrypt it thus knowing he is the winner and can communicate with the seller to further buy the auctioned good.

**5 Security Analysis and Evaluation**

Security of the proposed model ARPAN will be judged on the properties that are necessary for the auction and we will evolution of the proposed model is holding against such properties.

**5.1 Privacy**

In this subsection we provide the proof that privacy of the bids of bidders is safe from the auctioneer. As bids are padded by the random number of two bidders the original information of the bid by auctioneer is not possible without knowing the random numbers of the two bidders. Thus, we can claim that the bids are safe till two bidders don’t share their random numbers to the auctioneer.

**5.2 Accountability**

Accountability is holding because no bidder can submit fake deceitful bids and diverge the auction from real path. Algorithm 3 of ambiguity removal phase helps the auctioneer to exactly know whether or not any such deceitful fake bid has been submitted by any bidder in between.

**5.3 Integrity**

As the time slots have been allotted in each phase and each phase allows only those bidders who have registered for the auction thus the integrity of the auction is maintained.

**5.4 non-repudiation**

The bidders are not allowed to change the bids that they cast once, in bid submission phase they cast their bids by generating secure two-party comparison pairs, is some bidder tries to cast multiple bids in form of secure two arty comparison pairs than the auctioneer can easily know such maliciousness through the help of algorithm 3 for ambiguity detection phase.

**5.5 Verifiability**

Verifiability of the auction result that all major previous PPA were lacking and we have addresses it in our work. Algorithm can be followed by any of the interested authority and can verify the auction result thus guarantying that no malicious behaviors’ have been shown and has gone unnoticed. The winner of the auction is the highest bidder only.

**5.7 Fairness**

By watching the working of all phases, the proposed auction schemes can be seen as fairness assuring in each phase each bidders gets equal set of opportunities and in the end no biasness is shown in the declaring of the winner thus, we can say that fairness is greatly achieved in proposed privacy preserving auction.

**5.8 Transparency**

**5.9 Anti-collusion**

Phase by phase structure of our proposed scheme guarantees that no bidder-to-bidder collusion will go unnoticed and the proposed form of auction will hold the anticollision property.

**6 Threat Model**

This section will analyze the threat model and confidently identify possible threats that our proposed privacy-preserving scheme can effortlessly tackle, providing a foolproof solution.

**6.1 Malicious Bidder**

This threat occurs when a bidder is malicious and may get deviated by the auction rules. The maliciousness that it can show is by submitting fake random numbers either to change the auction results or may be intending to know other bidders bid value. But in both cases our model will identify such threats with the help of ambiguity removal phases and such maliciousness will be detected before the result declaration of the auction.

**6.2 Malicious Auctioneer**

Malicious auctioneer may want to know the bid values of the bidder for its own profit but as in our model we are using paillier homomorphic encryption and random padding to the bid value. And the problem of finding the addends of large number is computationally hard it is thus guaranteed the auctioneer cannot find the actual bid values of the bidders.

**6.3 Bidders collusion**

In the case of the bidder’s collusion two or more bidder try to collude to change the action results or may try to know other bidders bid vales but this can be easily detected with the help of algorithm 3 till the number of dishonest bidders is 2. And they cannot find the other bidders bid values because every time a bidder encounters the bid value of other bidders either it is encrypted with the auctioneer’s public key or is padded with the random number of multiple participants.

**6.3 Corrupt Certifier**

Another threat that can comes to picture in the PPA models is the dependency upon the central authority as is can result in as single point of failure but in our proposed model no confidential data is shared to the certifier is a unique ID of the interested participants is shared thus no bid values can be extracted by the auctioneer or central authority even if they both collude.

Result is collusion free

**7Conclusion**

**Case Study**

In this case study we are going to give proof of the algorithm 3 that it will be working even if a single bidder submits false entry or even if two bidders collude and try to change the results of auction.

Let us consider the case of 3 bidders where each bidder submits the encrypted bid to auctioneer and the bids that will be received to auctioneer after the bid submission phase are

In this section we are going to provide the proof for algorithm 3.

**Statement 1:** “*even if a single bidder violates the protocol than it will not go unnoticed”*.

Let us consider bidders and Let’s say where each bidder bids respectively.

In the bid submission phase for bidder the bid pairs that will be generated and submitted to auctioneer in respect to all other bidder for secure two party comparison will be Let’s suppose a bidder submits

different random numbers for two shared messages for the purpose to create ambiguity in the auction results and let the different message be If the results of the auctions were only based on the bid submission phase than this ambiguity has gone unnoticed and had altered the auction results, but due to the bid reshuffling phase will again have to add random number to this message and after the completion of bid reshuffling phsae this packet will behaving the sum of random numbers of all bidders which will be . During the bid reshuffling phase, the auctioneer adds its random number to each packet hence can not identify the message in which he added the fake random number.

Now following the ambiguity removal protocol auctioneer will arrange the packets in following manner

--------------------------------------------------

-------------------------------------------------

. . .

. . .

. . .

. . .

-----------------------------------------------

As the sum of all these rows must be same which is where the value of ,but as if bidder has submitted fake random value the last row will yield a different sum which is and this sum will be different from all other rows thus if a single row yields different result than it’s a red flag for auctioneer to discard the auctioneer.

**Statement 2: “***If two or more bidders try to collude and disrupt the auction results than also the ambiguity will not go unnoticed”*

This ambiguity detection algorithm can be applied to the data collected from bid submission phase and ambiguity can be detected for a single bidder case but the algorithm fails if more than one bidder are trying to collude to make the result ambiguous. Two or more bidders that are performing collusion can easily by pass the ambiguity detection by faking the random number in such differences so that the sum of rows comes exactly same. For example, let say the bidders are colluding and they have faked random number as shown in equation .

(6)

Now the auctioneer arranges the bids of bid submission phase in following manner.

--------------------------------------------------

--------------------------------------------------

. . .

. . .

. . .

. . .

-----------------------------------------------

Now in this case the sum of each row will yield same sum and auctioneer will be convinced that auction results are not ambiguous but in reality, two bidders have cleverly colluded. Due to this weak point, we have proposed bid reshuffling phase and identity of the padded bids where the bidder have added fake random numbers will not be disclosed thus the arrangement of the bids by the auctioneer will be

--------------------------------------------------

-------------------------------------------------

. . .

. . .

. . .

. . .

----------------------------------------------

In this condition also same as statement one last row will yield a different sum thus collusion will be easily detected.

Such collusion can be detected till n-2 bidders as dishonest because till than anonymity will be maintained of the submitted bids but if n-1 bidders collude than in that case only a single bidder will be honest and other bidders can easily identify that this one is the honest bidder submitted value of the bid that they have got for their random number addition and in this case they submit their honest value but in all other cases they follow the equation 6 and they will bypass the ambiguity check. So, our proposed algorithm is full proof till n-2 bidders are dishonest.