Scientific Research Publishing

Int. J. Communications, Network and System Sciences, 2022, 15, 94-110
https://www.scirp.org/journal/ijcns
ISSN Online: 1913-3723
ISSN Print: 1913-3715

Design of a Peer-to-Peer Energy Trading Platform Using Multilayered Semi-Permissioned Blockchain

Ishtiaque Zaman, Md Mahmudul Hasan, Miao He, Michael G. Giesselmann

Department of Electrical and Computer Engineering, Texas Tech University, Lubbock, USA
Email: ishtiaque.zaman@ttu.edu, md-mahmudul.hasan@ttu.edu, miao.he@ttu.edu, michael.giesselmann@ttu.edu

How to cite this paper: Zaman, I., Hasan, M.M., He, M. and Giesselmann, M.G. (2022) Design of a Peer-to-Peer Energy Trading Platform Using Multilayered Semi-Permissioned Blockchain. *Int. J. Communications, Network and System Sciences*, **15**, 94-110.
https://doi.org/10.4236/ijcns.2022.157008

Received: June 19, 2022
Accepted: July 26, 2022
Published: July 29, 2022

Abstract

A secured and scalable Peer-to-Peer (P2P) energy trading platform can facilitate the integration of renewable energy and thus contribute to building sustainable energy infrastructure. The decentralized architecture of blockchain makes it a befitting candidate to actualize an efficient P2P energy trading market. However, for a sustainable and dynamic blockchain-based P2P energy trading platform, few critical aspects such as security, privacy and scalability need to be addressed with high priority. This paper proposes a blockchain-based solution for energy trading among the consumers which ensures the systems' security, protects users' privacy, and improves the overall scalability. More specifically, we develop a multilayered semi-permissioned blockchain-based platform to facilitate energy transactions. The practical byzantine fault tolerant algorithm is employed as the underlying consensus for verification and validation of transactions which ensures the system's tolerance against internal error and malicious attacks. Additionally, we introduce the idea of quality of transaction (QoT)—a reward system for the participants of the network that eventually helps determine the participant's eligibility for future transactions. The resiliency of the framework against the transaction malleability attack is demonstrated with two uses cases. Finally, a qualitative analysis is presented to indicate the system's usefulness in improving the overall security, privacy, and scalability of the network.

Keywords

Blockchain, Peer-to-Peer Energy Trading, Distributed Energy Resources, Consensus

1. Introduction

The adoption of distributed energy resources (DERs) is increasingly being pro-

moted worldwide to address the ongoing energy crisis as well as to reduce the dependency on fossil fuels for energy generation [1]. Accordingly, with the rise of the DERs, the microgeneration of renewable energy is also increasing. As a result, consumers are nowadays also gaining the capability to generate energy and thus, transforming into prosumers. However, these prosumers need to be motivated to maintain the continuity of generation as well as to facilitate the integration of renewable energy into the power grid. Prosumers can trade energy among each other to incentivize themselves and earn revenue. Moreover, this potential to trade among each other unfolds a research prospect that can eventually increase renewable energy generation capacity. The sporadic generation of renewable energy can cause grid instability that can be addressed by an efficient P2P energy trading platform. The balance between demand and generation of energy can be achieved, and thus a self-sustainable energy infrastructure can be built with the help of a comprehensive P2P platform. Therefore, a P2P trading platform can be an effective solution for DER integration [2] considering the financial interest of the prosumers as well as the practical viability for implementation.

The decentralized landscape of DER completely matches the underlying distributed infrastructure of blockchain [3]. Moreover, blockchain maintains information transparency and symmetry through shared and immutable ledgers, which are distributed among the participants [4]. An underlying consensus algorithm protects the network to ensure trusted, or trust-less collaboration among the participants [5]. The distributed architecture, secured transaction capability, and transparency in information sharing make it an ideal candidate for realizing a P2P energy trading platform in smart grid. A blockchain-based platform was developed and implemented on the Alternating Direction Method of Multipliers algorithm [6]. Reference [7] developed a proof-of-work (PoW) based blockchain for enabling cloud based manufacturing services. An energy trading framework using a Bitcoin-based payment system and PoW [8] consensus algorithm was developed in [9]. Reference [10] developed an auction mechanism for energy trading market using the Ethereum blockchain which was also backed by PoW consensus. However, in PoW, consensus among the network participants is achieved by solving complex cryptographic puzzles which in turn requires high computation power and more bandwidth. Reference [11] proposed blockchain-based platform using the principles of proof-of-stake (PoS) [12] consensus. However, PoS based blockchain becomes more centralized with the increase of the network participants, resulting in making the system vulnerable to a single point of failure (SPoF) attack. A private blockchain was developed by [13] where a digital currency named NRGcoin was introduced and prosumers were encouraged to insert their locally generated energy into the grid. The low-voltage distribution system operator (DSO) was responsible for determining the value of the NRGcoin based on the smart meter data of the prosumers. Intuitively, this method can be exposed to SPoF attack and potentially compromise

the privacy of the prosumers. A private blockchain-based framework was introduced by [14] using Redundant Byzantine Fault Tolerant (RBFT) consensus method [15] where energy is crowdsourced in a trading marketplace. However, a private blockchain is managed by a central authority that can be vulnerable to SPoF attacks.

Blockchain is increasingly being adopted by the grid and utilities to facilitate energy trading among their consumers through pilot projects. A pilot project named Brooklyn Microgrid was launched by LO3 Energy [16] where an energy marketplace was developed to facilitate trading of solar energy among the network participants. Another company named Power Ledger [17] conducts several pilot projects to provide P2P energy trading platform to their consumers who have solar energy generation facility.

Although the feasibility of blockchain for P2P energy trading is proved by the contemporary solutions and the pilot projects, blockchain has some inherent limitations that are crucial for a successful energy transaction. The underlying structure of blockchain varies depending on different consensus mechanisms, and the implementation of these algorithms is problem-specific [18]. The most commonly used consensus algorithms for blockchain are PoW and PoS due to their favorable outcomes in various applications, including financial transactions. However, applying these algorithms require few assumptions which potentially limit a generalized blockchain-based implementation for P2P energy trading. PoW and PoS are applied to develop public blockchains where users are completely anonymous (e.g., Bitcoin, Ethereum, etc.). However, complete anonymity in case of an energy infrastructure can potentially cause privacy violation of the prosumers [19]. On the other hand, a private blockchain can be a better alternative. However, a private blockchain is more centralized than a public blockchain which makes the system vulnerable to attackers [20]. When deciding on a comprehensive platform for P2P energy trading, the following aspects need to be considered:

- ***Security***: As a crucial commodity, energy infrastructure must be secured from potential malicious attacks. Moreover, in P2P energy trading, the end-users (prosumers) carry out the energy transactions instead of skilled system engineers. This can initiate internal errors that may extend to system failure. Therefore, a fault-tolerant energy trading platform is necessary that is resilient from external attacks as well as operational failures.
- ***Privacy***: One of the key features of blockchain is the transparency of information among its participants. In a public blockchain, these participant's identities are anonymous, and in a private blockchain, the identities are managed by a central authority. In a P2P energy trading scenario, the complete anonymity of the prosumers may lead to privacy violations since all the network participants share the smart meter data. Therefore, a platform that can automatically control the anonymity of the prosumers is necessary to avoid a potential privacy breach.

- ***Scalability***: Blockchain-based on PoW requires high computation power to validate transactions, which makes the system energy inefficient and limits its scalability. Therefore, a scalable P2P energy trading platform is necessary to integrate a large number of prosumers and execute transactions at a high rate.

In this paper, we present a multilayered semi-permissioned blockchain-based P2P energy trading platform considering the aforementioned aspects for a sustainable P2P energy market. The term "semi-permissioned" refers to the fact that identity of a new prosumer is managed by a trusted authority in the primary stage to perform the verification task. However, no trusted authority is required during the transaction validation process. Practical Byzantine Fault Tolerant (PBFT) [21] is used as the underlying consensus protocol to implement the blockchain network. Additionally, a reward system—Q-score, is introduced to measure the quality of a transaction (QoT) that eventually helps determine a prosumer's eligibility to participate in future transactions. Part of the work was presented in our conference paper [22], and this paper extends the work with specific implementation and technical details for developing the platform from scratch and to conduct use case studies.

The rest of the paper is structured as follows: Section 2 presents the proposed framework, including a brief background of the consensus mechanism and a detailed explanation of various framework components. In Section 3, we present the implementation details of the blockchain-based framework. The resiliency of the proposed framework against the transaction malleability attack is demonstrated with two use cases in Section 4. In Section 5, a qualitative analysis is presented to indicate the system's usefulness in improving the security, privacy, and scalability of the network. Finally, Section 6 concludes the article.

2. Proposed Framework

The proposed framework employs the PBFT consensus method to realize a blockchain-based solution for verification of new prosumers and validation of transactions. The smart meters of the network participants are used as nodes in the network to implement the PBFT consensus. We present a brief background of the PBFT consensus in the next subsection. The overall system can be viewed as a multilayered structure where each layer is responsible for different tasks. The tasks include initiating transaction, generating smart contract, implementing consensus algorithm, transferring data in terms of distributed ledger and control signal, and performing electronic fund and energy transfer. The later subsection describes the structure of the proposed framework along with a detailed explanation of each of the components of the framework.

2.1. PBFT Consensus

Figure 1 illustrates the mechanism of PBFT consensus. This consensus has three types of nodes: client, leader, and replica nodes. The client node invokes the

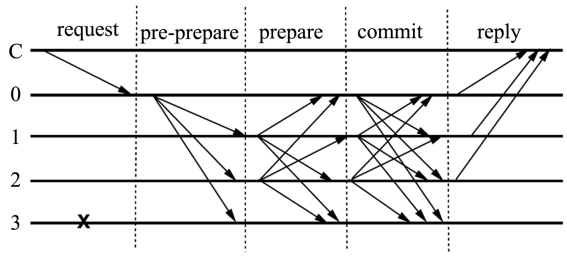

Figure 1. PBFT consensus mechanism.

transaction request. The leader node is responsible for verifying the request and encapsulating the transactions into blocks. All the other nodes in the network except the client and the leader node are called replica nodes. The whole consensus is achieved through several stages. The first stage is the request stage, where the client node, C, sends a *request* type message to the leader to initiate a transaction. In the pre-prepare stage, the leader receives the request from the client and broadcasts a *pre-prepare* type message to all the replica nodes. The replica nodes check the validity of the message through a public key signature. In the prepare stage, each replica node multicasts a *prepare* type message to all other replica nodes (including the leader node) and adds both the *pre-prepare* and *prepare* messages to its local log. A node accepts and adds the *prepare* message to its log only when the information matches the *pre-prepare* message. After the prepare stage, the commit stage starts where all replica nodes multicast a *commit* type message among each other. The validation process of the messages is the same as that of the previous stage. The last stage of the consensus is the reply stage, where all nodes send a *reply* type message to the client. The client waits for a minimum of $f + 1$ valid replies from different nodes before finally accepting the result of the operation. Here, f represents the maximum number of nodes that may be faulty. This makes the consensus more fault-tolerant because the system remains operational even if there is f number of faulty nodes when the client receives at least one more than f valid messages.

2.2. Architecture of the Proposed Framework

Figure 2 illustrates the multilayered structure of the framework. The communication and negotiation on a transaction between two prosumers take place in the application layer. Also, transactions are initialized at this stage. Upon initialization of a transaction, a smart contract is generated, which is validated and executed in the blockchain layer. Ledgers of all the network participants are shared in the network layer. The physical layer is responsible for energy transfer. We consider the grid to be bi-directional with net metering arrangements [23] to facilitate energy trading. A two-stage blockchain-based framework is proposed in this paper. The verification of a new prosumer takes place in the first stage, and the second stage is used transaction validation. Additionally, the quality of a transaction is measured by a reward value-Q-score, to rate a participant in a transaction. The overall structure of the framework is demonstrated in Figure 3.

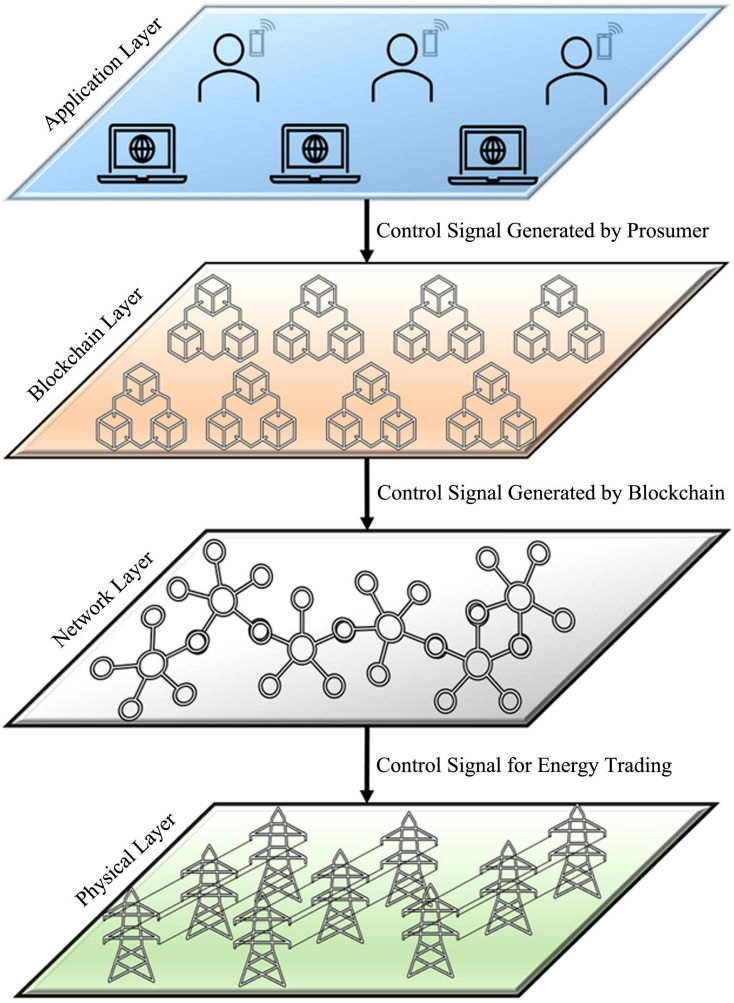

Figure 2. Multilayer model for energy trading.

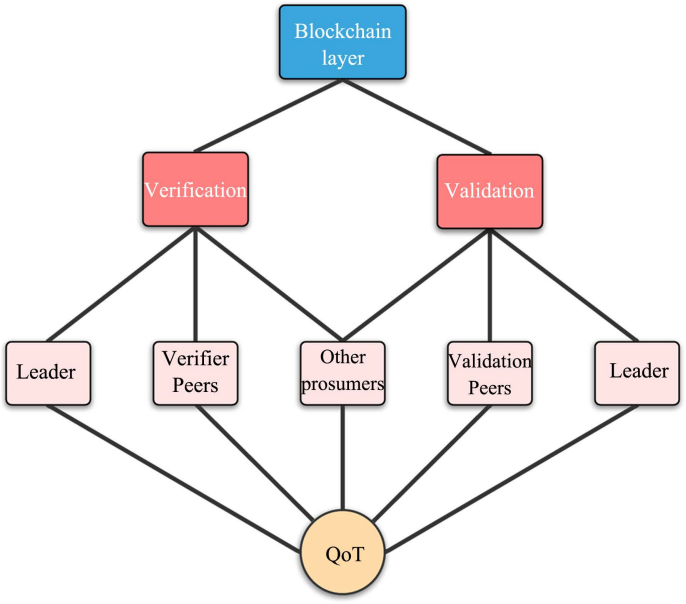

Figure 3. Framework components.

2.2.1. Verification of New Prosumers

To include a new prosumer in the network, the identity of the new prosumer must be verified. This verification is not necessary in the case of public blockchains since the participants are anonymous. However, in private blockchains, a trusted third party is required to manage the identity of the prosumers and to perform the verification. The requirement of a third party is avoided in [24] by implementing "Certificate of Existence". However, this assumes that the smart meters populate key pairs and act as the certification authority. The proposed framework uses the network's consensus among the existing prosumers to perform the verification. The idea is to verify a new prosumer with the help of the existing participants instead of a certified authority.

Verification of a new prosumer to join the network is performed by a set of verifier nodes. Each n ode represents the smart meter of each household. Clustered sampling method is used to select the verifier nodes. Verifier nodes are the prosumers of the network that voluntarily offer their reputation (Q-score) at stake to obtain the eligibility for validating nodes or creating new blocks and earn credits in the form of Q-scores. Upon selection of the verifier nodes, the reputations of these members become visible to the network. X509 certificates are used to verify new nodes. Additionally, the verifier set is changed after verifying a new node. A leader node creates a new entry to the block and includes the identity of the new node in the entry. A leader node l_i at I^{th} time instant t_i is determined by the following:

$$l_i = \left[\left(t_i - t_0\right)/\alpha\right]\%\eta \tag{1}$$

where t_0 is the time instant of the first block, α is the time interval between two blocks, and η is the total number of nodes at I^{th} time. A new leader is selected for each verification according to Equation (1). The complete workflow of the verification is presented in **Figure 4**. At first, a leader node receives the joining request, creates a verification request, adds the request to a block, or creates a new block depending on the block's current size. At this stage, the leader node only accepts a request when the identity of the new node matches with identity provided by the certified authority. The leader node then sends the request to the other replica nodes in the verifier set. The replica nodes verify the following conditions:

Condition 1: The block was generated from the leader at t_i.

Condition 2: No other blocks have been generated from the same leader at t_i.

If the both the conditions are passed, the replica nodes then send the request to the leader node. Finally, the leader node executes the request and adds the new node to the network.

2.2.2. Validation of Transactions

Once the prosumer initiates a transaction, a smart contract is generated and is validated by the other prosumers of the network. The generated smart contract determines a prosumer's role in a transaction as a seller or buyer. Each node

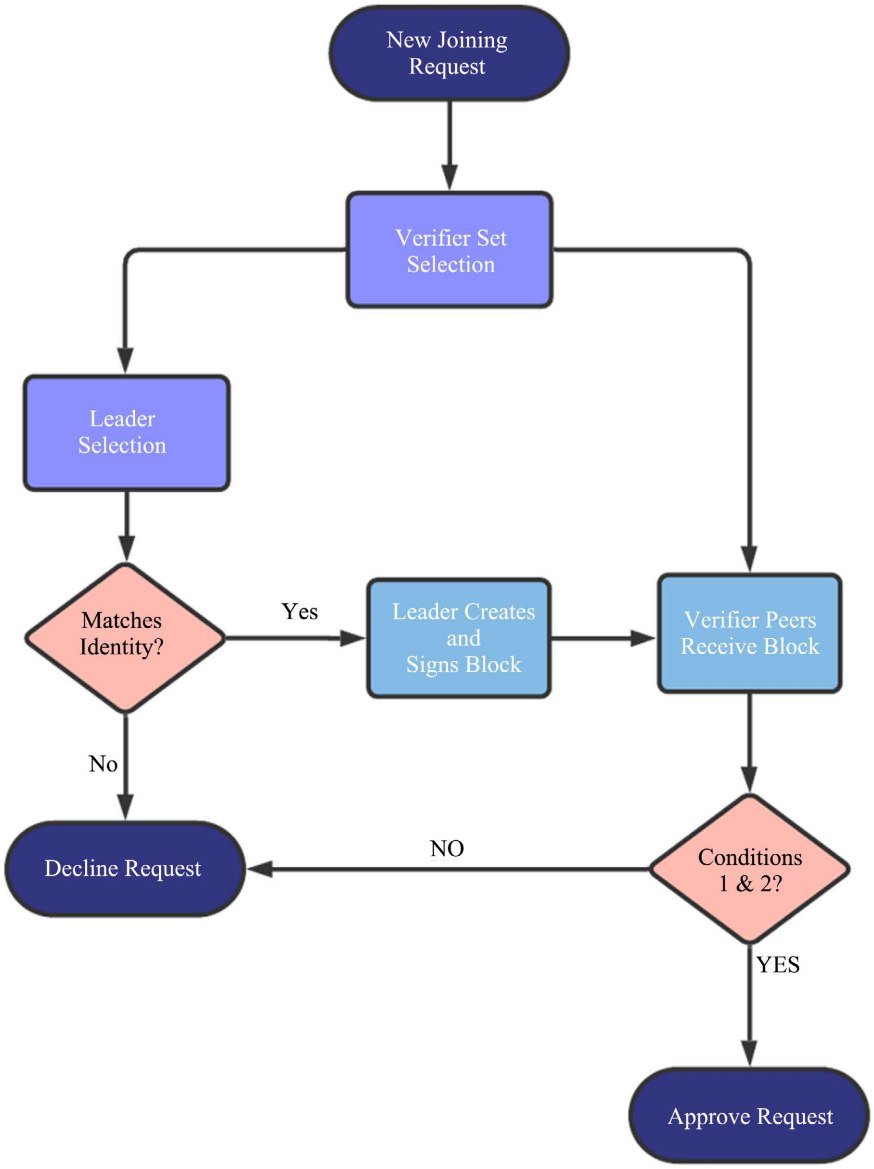

Figure 4. Verification of new prosumer.

contains a ledger that enlists two data types: 1) Transaction log and 2) State. All the transactions can be tracked by Transaction log which is an immutable data type. State is a key-value pair which is versioned and contains energy and wallet data. Information stored in the state data types can be added but not removed or edited. The information in states includes the energy and wallet amount, smart meter ID, etc. The validation workflow is presented in **Figure 5**. There are three sub-stages in transaction validation process:

Preparation stage: A transaction is initiated based on the agreement of two prosumers. Clustered sampling method is used to determine the set of validator nodes and a leader is selected using Equation (1) from the validator peers. The leader node acts as an intermediate or relay node for a transaction. For each transaction validation task, a new cluster is selected, and the leader is also altered,

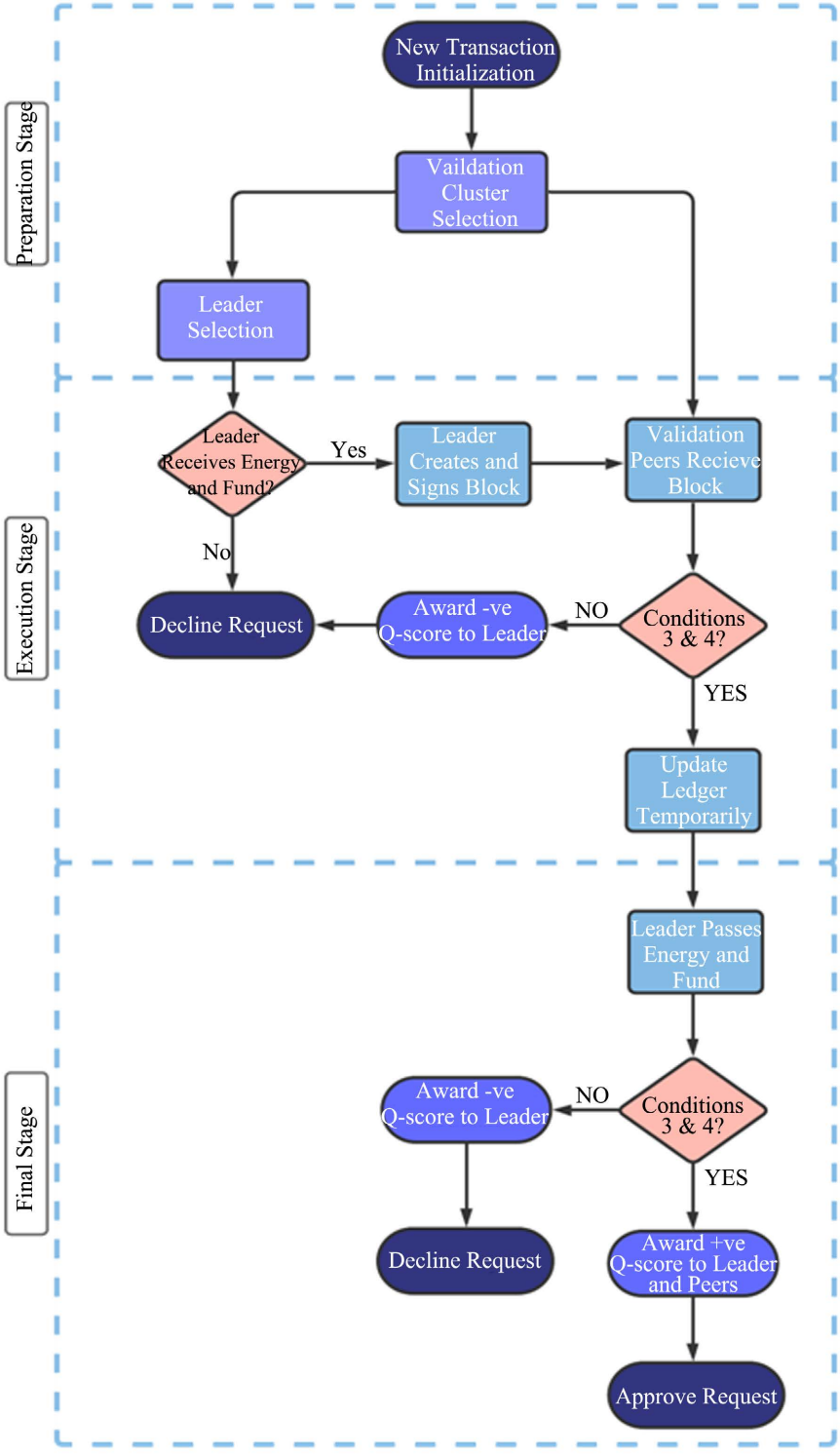

Figure 5. Validation of transaction.

making the process less susceptible to SPoF attacks. A smart contract is invoked for each transaction initiation. The smart contract consists of transaction related information, *i.e.*, available energy, energy price, smart meter IDs of the participating nodes such as seller, buyer, leader, and validator nodes.

Execution Stage: At the execution stage, the leader node receives the transaction request and the energy amount, updates its energy and wallet information as per the smart contract and sends the transaction request to the replica nodes for validation. The validator nodes check the following conditions:

Condition 1: The energy information of seller and leader are updated according to the smart contract.

Condition 2: The wallet amounts of buyers and leader are updated according to the smart contract.

The transaction reaches the final stage provided that both the conditions are passed, and the ledger is updated temporarily. Here, the term "temporarily" means that only the validator nodes and the leader update their ledger, whereas the other network member's ledger is not updated yet.

Final Stage: In this stage, the leader node relays the energy received in the execution stage to the buyer and updates the wallet information accordingly. The newly updated ledger is also shared with the validator nodes to check the following conditions:

Condition 3: The energy amounts of buyer and leader are updated according to the smart contract.

Condition 4: The wallet amounts of seller and leader are updated according to the smart contract.

The transaction is complete when both the condition pass and the ledger are updated permanently. Here, the term "permanently" refers to the fact that all the network members now have the updated version of the ledger that includes the transaction details.

2.2.3. Quality of Transaction (QoT)

A reward point is assigned to the corresponding nodes upon completion of each transaction. This variable is denoted as Q-score. Q-score serves as a tool to determine the quality of a transaction and rate the participants involved in a transaction. A successful transaction results in a positive Q-score for the seller and buyer of that particular transaction. The corresponding leader, verifier nodes, and validator nodes involved in a successful transaction are also rewarded with positive Q-scores. Contrarily, one of the two cases occurs when a transaction is declined. Either the seller fails to deliver the energy, or the buyer fails to pay the price according to the smart contract. When the seller is at fault, a negative Q-score is assigned to penalize the seller and vice versa when the buyer is at fault. The total Q-score in i^{th} transaction is determined by:

$$\left(\text{Q-score}\right)_{(i)} = v_{(i)} + L_{(i)} + f\left(x_{(i)}, \alpha\right) \tag{2}$$

where,

$$v_{(i)} = v_{(i-1)} * x_{(i)}$$

$$L_{(i)} = L_{(i-1)} * x_{(i)}^{3},$$

and

$$f(x,\alpha)=\begin{cases}(S+B)*x^5 & \text{if } x>0\\ \alpha*S*x^5 & \text{if } x<0\,\&\,\alpha=+1\\ \alpha*B*x^5 & \text{if } x<0\,\&\,\alpha=-1\end{cases}$$

where, $x_{(i)}$ is the Q-score of i^{th} transaction and $x_{(i)}\in R$. S and B are the Q-scores of seller and buyer, respectively. $L_{(i)}$ and $v_{(i)}$ are the Q-scores of the leader and verifier/validator nodes, respectively. α is a signed coefficient to represent the quality of the i^{th} transaction when a transaction is declined. $\alpha=+1$, when the seller fails to deliver the energy and $\alpha=-1$, when the buyer fails to pay the price. The total Q-score of a transaction is added to the ledger and is shared among all the participating nodes. The Q-score determines the future trading capability of a node. More specifically, a higher Q-score of a node ensures the credibility of the corresponding prosumer for future transactions. In contrast, a lower Q-score affects the likeliness of the corresponding prosumer's future transactions.

3. Implementation

This section presents the implementation details of the blockchain framework. Hyperledger Composer framework [25] has been used to implement the blockchain network. The first step in developing the blockchain network is to define the network components. Network components include participants, assets, and trade. In the case of participants, a class is declared to define the attributes related to a participant. These attributes are Participant ID, Contact, and Participant Type (Buyer/Seller/Verifier/Validator, etc.) In the P2P energy trading scenario, energy can be considered an asset. Therefore, an energy class is defined to include the attributes related to energy, such as the amount of energy, energy availability, energy source type, energy generation time stamps, etc. are defined under energy class. Another component is trade, for which the class is declared as *org.framework.peer2peer.trade*. The attributes from the energy and participants class are imported in this class. The attributes of this class are tradeID and energy route (origin and destination smart meterID). Additionally, transactions are defined in this class along with the event that is emitted from the transactions. **Figure 6** shows different attributes of the participants, energy, and trade classes. These codes are written in Hyperledger Composer modeling language, which is an object-oriented language for defining the domain model of a business network application. After defining the components, Hyperledger composer playground is used as a web-based application to simulate and test different scenarios. The playground provides a user interface for a business network's configuration, deployment, and testing. This application is uploaded or deployed in the Hyperledger composer playground for simulation. Using playground, new participants are added to the network. When a participant is added to the network, a participant ID is automatically generated, and other attributes such as smart meter id, name, address, email, etc., are provided. Also, the type of the participant is specified, such as buyer, seller, verifier, leader, validator, etc.

```
namespace org.framework.peer2peer.participant

abstract participant p2pParticipant identified by participantId{
  o String           participantId
  o Identity         key
  o Contact          contact
}


concept Identity {
  o String prosumerID
  o String smartmeterID
}
concept Contact {
  o String fName
  o String lName
  o String address
  o String email
  o String cellNo
}

enum ParticipantType {
  o buyer
  o seller
  o leaderVerifier
  o verifierpeer
  o leaderVlidator
  o validationpeer
}
```

(a)

```
namespace org.framework.peer2peer.energy

asset Energy identified by packetId {
  o String        packetId
  o Double        amount
  o Identity      id
  o Contact       contact            optional
  o DateTime      generationDatetime
  o Boolean       available
  o Boolean       claimed
  o energySource  energyType         optional
}

concept Identity {
  o String prosumerID
  o String smartmeterID
}
concept Contact {
  o String fName
  o String lName
  o String address
  o String email
  o String cellNo
}

enum energySource {
  o   solar
  o   wind
  o   fossil
}
```

(b)

```
namespace org.framework.peer2peer.trade

import org.framework.peer2peer.energy.Energy

import org.framework.peer2peer.participant.p2pParticipant

asset Trade identified by tradeId {
  o String                tradeId
  o Route                 route
  --> Energy              energypack
  --> p2pParticipant      participantId
}


concept Route {
  o String      origin
  o String      destination
  o DateTime    schedule
}

transaction createTrade {
  o String          tradeKey
  o Route           route
}

event tradeCreated {
  o String keyTrade
}
```

(c)

Figure 6. Defining attributes of (a) Participants, (b) Assets, and (c) Trades.

(a) (b) (c)

Figure 7. (a) Buyer, (b) Seller and (c) Validation peers on playground.

Figure 7 shows the assigned buyer, seller, and validators in playground. Energy is added as assets in the network which requires specifying the attributes related to energy assets, such as the amount of energy, energy source id, availability, etc. Also, attributes such as energy packetID and Time Stamps are auto generated. In the case of trade, tradeID and timestamps are auto generated. However, the trade route that includes origin and destination are specified. From the energy class, the energy id for trading is imported. Also, the participants relevant to this trade are specified by their respective IDs. The submit transaction option in the playground allows creating and executing a transaction based on the smart contract developed in JavaScript. The smart contract contains the logic for executing a transaction. The composer library functions are used in this script. Additionally, functions are developed to implement and execute the transaction scenarios. Below are some of the functions that have been used to execute a transaction:

createTrade()—This function takes trade id and route as input argument. It checks the availability of the energy amount in a transaction.

generateTradeId()—This function is used to generate the tradeID and timestamps.

AssignEnegy()—This function creates a transaction and updates the energy, participant, and trade information after a transaction.

4. Use Cases

In this section, two possible attack scenarios are presented to demonstrate the resiliency of the proposed framework against potential transaction malleability attack. The first scenario represents a malicious seller who violates the smart contract by not delivering the agreed amount of energy but intends to receive the price of the energy. The second scenario portrays a malicious buyer who violates the smart contract by not paying the price of the agreed amount of energy but intends to receive the energy. Our proposed framework is able to detect the fraudulent party in both cases and decline the transaction. **Figure 8(a)** and **Figure 8(b)** illustrate the two attack scenarios and describes how the malicious party is identified and the transaction is declined.

Malicious Seller Scenario: In this scenario, an attacker representing himself as a seller initiates a transaction with a genuine buyer. This scenario depicts a potential system failure or a malicious attack that can lead to double-spending. In this scenario, the attacker manages to change the transaction details in the smart contract. More specifically, the attacker may change the energy value in the generated smart contract. The intention is to make the buyer believe that the transaction was unsuccessful and motivate the buyer to send the money again. The attacker sends out the altered transaction before the original transaction, and the transaction gets declined naturally. The buyer then sends the money again to ensure integrity which causes double spending for the buyer. In the proposed framework, the leader receives a smart contract that includes the altered energy value and sends the transaction request to the replica nodes after temporarily updating his ledger. However, since the change in the energy amount of the seller does not match with the received smart contract, the validation

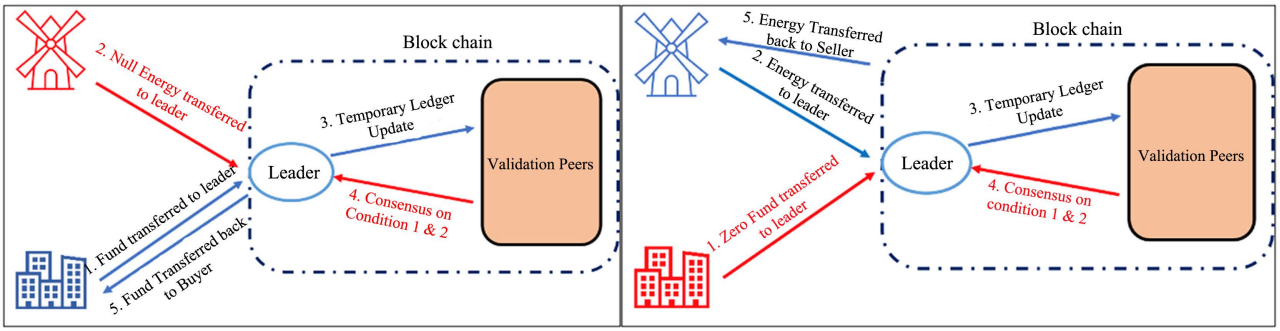

Figure 8. (a) Malicious seller scenario, (b) Malicious buyer scenario.

peers send out the reply messages to the buyer indicating the mismatch in the ledger. Since most of the buyer's reply messages indicate the mismatch, the transaction is declined at the execution stage. Here, the leader's role prevents the double spending of the buyer.

Malicious Buyer Scenario: In this case, an attacker representing himself as a buyer may also initiate a transaction with a genuine seller. This is another case of a transaction malleability attack where the attacker manages to change the amount of energy in the generated smart contract. In a transaction malleability attack, the attacker broadcasts the changed transaction before the original transaction. The intention is to cause the double spending of the seller. Since the wrong transaction is sent out before the original transaction, the transaction is declined, and the seller sends the energy again, which causes double spending for the seller. However, in the case of the proposed framework, at first, the leader receives the smart contract, including the changed amount, and sends out the transaction request to the replica nodes by temporarily updating his ledger. However, the validation peer's ledger for this transaction does not match that of the seller or leader. The replica nodes then send the reply messages to the buyer indicating the mismatch in the ledger. Since the majority of the reply messages received by the buyer indicate the mismatch, the transaction is declined at the execution stage. Here, the leader's role prevents the double spending of the seller.

5. Security, Privacy and Scalability

In this section, we present a qualitative analysis of the security, privacy and scalability issues pertaining to an energy trading platform.

5.1. Security

In a blockchain network, all the transactions and ledgers are protected by certificates to ensure security. A trusted central authority is responsible for these certificates in a private blockchain. As an initiative to make the system less centralized, the proposed framework utilizes the idea of a two-stage blockchain. We considered the following common attack vectors for a qualitative analysis of the proposed framework:

Denial-of-service (DoS) attacks: A DoS attack typically involves a simultaneous occurrence of a large number of transactions in a network to occupy the network resources *i.e.*, bandwidth with an aim to obstruct the normal operation or to delay the transactions. The proposed framework prevents this attack by 1) using different verifier nodes for different verification tasks, 2) using different validator nodes for different validations tasks, 3) assigning different roles among the network participants, and 4) Using Q-score to rate the transactions as well as the participating nodes.

51% attack: PoW-based blockchain can be compromised by gaining 51% computation power of the whole network. More specifically, if an attacker pos-

sesses 51% computation power of the network, then the network is compromised. However, an attacker is required to gain control over 51% of the nodes to initiate such attacks in the proposed framework. Intuitively, an attacker can realize 51% computation capability even from one node. Thus, the system is more resilient against 51% attack since the possibility of compromising 51% of the nodes is very low comparing to the possibility of gaining 51% computation power.

5.2. Privacy

In a private blockchain, the identities of the participants are verified by a trusted certification authority. Therefore, a private blockchain is vulnerable to SPoF attacks. Moreover, the privacy of the users is compromised to the certification authority. On the other hand, no identity verification is required in a public blockchain. However, privacy can be violated in a public blockchain by identifying the activity pattern of a particular participant. Therefore, an appropriate identity verification method is important for a privacy sensitive application such as energy trading. The proposed framework preserves the privacy of its prosumers by implementing different stages for verification and validation. In the verification stage, the identity of the new node is readable to only the verifier node, which is protected by X509 certificates. During the validation stage, instead of the identities, only the Q-scores of the corresponding prosumers are shared in the ledger.

5.3. Scalability

For a sustainable blockchain based marketplace, a scalable network is necessary to accommodate a large number of participants and transactions. In PoW, each transaction validation requires resource intensive mining which potentially limits the large-scale implementation of a blockchain network. However, in the proposed framework, both the verification and validation stage involve only a number of nodes selected using a clustered sampling method. Also, recall that, in this framework, only the leader is responsible for executing a transaction, whereas other replica nodes are responsible for validation or verification tasks. This avoids the requirement of resource-consuming computations for transaction validation tasks and consequently improves the system's overall scalability.

6. Conclusion

This paper proposes a semi-permissioned blockchain-based P2P energy trading platform. PBFT consensus algorithm is employed as the underlying consensus for new prosumer verification and transaction validation. The decoupling of verification and validation tasks makes preserves the privacy of the participants. The overall scalability of the system is improved by avoiding the resource-consuming validation process. The alteration of the verifier and validator nodes increases the overall resiliency of the platform and enhances the system's privacy. The

future scope of this work involves implementing the proposed platform in an islanded microgrid for comprehensive performance comparison against state-of-the-art solutions.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Hess, D.J. and Gentry, H. (2019) 100% Renewable Energy Policies in US Cities: Strategies, Recommendations, and Implementation Challenges. *Sustainability: Science, Practice and Policy*, **15**, 45-61. https://doi.org/10.1080/15487733.2019.1665841

[2] Morstyn, T., Teytelboym, A., Hepburn, C. and McCulloch, M.D. (2019) Integrating P2P Energy Trading with Probabilistic Distribution Locational Marginal Pricing. *IEEE Transactions on Smart Grid*, **11**, 3095-3106. https://doi.org/10.1109/TSG.2019.2963238

[3] Burger, C., Jens, W., Andreas, K. and Philipp, R. (2016) Blockchain in the Energy Transition. A Survey among Decision-Makers in the German Energy Industry. *DENA German Energy Agency*, **60**, 1-44.

[4] Mainelli, M. and Smith, M. (2015) Sharing Ledgers for Sharing Economies: An Exploration of Mutual Distributed Ledgers (Aka Blockchain Technology). *Journal of Financial Perspectives*, **3**.

[5] Beck, R., Czepluch, J., Lollike, N. and Malone, S. (2016) Blockchain-The Gateway to Trust-Free Cryptographic Transactions.

[6] Münsing, E., Mather, J. and Moura, S. (2017) Blockchains for Decentralized Optimization of Energy Resources in Microgrid Networks. 2017 *IEEE conference on control technology and applications* (*CCTA*), Maui, 27-30 August 2017, 2164-2171. https://doi.org/10.1109/CCTA.2017.8062773

[7] Bahga, A. and Madisetti, V.K. (2016) Blockchain Platform for Industrial Internet of Things. *Journal of Software Engineering and Applications*, **9**, 533-546. https://doi.org/10.4236/jsea.2016.910036

[8] Nakamoto, S. (2008) Bitcoin: A Peer-to-Peer Electronic Cash System. *Decentralized Business Review*, Artcile ID: 21260.

[9] Aitzhan, N.Z. and Svetinovic, D. (2016) Security and Privacy in Decentralized Energy Trading through Multi-Signatures, Blockchain and Anonymous Messaging Streams. *IEEE Transactions on Dependable and Secure Computing*, **15**, 840-852. https://doi.org/10.1109/TDSC.2016.2616861

[10] Hahn, A. *et al.* (2017) Smart Contract-Based Campus Demonstration of Decentralized Transactive Energy Auctions. 2017 *IEEE Power & Energy Society Innovative Smart Grid Technologies Conference* (*ISGT*), Washington, DC, 23-26 April 2017, 1-5. https://doi.org/10.1109/ISGT.2017.8086092

[11] Horta, J., Kofman, D. and Menga, D. (2017) Novel Paradigms for Advanced Distribution Grid Energy Management. arXiv: 1712.05841.

[12] King, S. and Nadal, S. (2012) Ppcoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake. Self-Published Paper, 19.

[13] Mihaylov, M., Jurado, S., Avellana N., *et al.* (2014) NRGcoin: Virtual Currency for

Trading of Renewable Energy in Smart Grids. 11*th International Conference on the European Energy Market* (*EEM*14), Krakow, 28-30 May 2014, 1-6. https://doi.org/10.1109/EEM.2014.6861213

[14] Wang, S., *et al.* (2019) Energy Crowdsourcing and Peer-to-Peer Energy Trading in Blockchain-Enabled Smart Grids. *IEEE Transactions on Systems*, *Man*, *and Cybernetics*: *Systems*, **49**, 1612-1623. https://doi.org/10.1109/TSMC.2019.2916565

[15] Aublin, P.-L., Mokhtar, S.B. and Quéma, V. (2013) Rbft: Redundant Byzantine Fault Tolerance. 2013 *IEEE* 33*rd International Conference on Distributed Computing Systems*, Philadelphia, 8-11 July 2013, 297-306. https://doi.org/10.1109/ICDCS.2013.53

[16] The Future of Energy: LO3 Pando: Blockchain, Transactive Grids, Microgrids, Energy Trading: LO3 Tokens and Information. https://lo3energy.com/

[17] Power Ledger: Energy Trading Platform. https://www.powerledger.io/platform

[18] Baliga, A. (2017) Understanding Blockchain Consensus Models. *Persistent*, **4**, 14.

[19] Varodayan, D. and Khisti, A. (2011) Smart Meter Privacy Using a Rechargeable Battery: Minimizing the Rate of Information Leakage. 2011 *IEEE International Conference on Acoustics, Speech and Signal Processing* (*ICASSP*), Prague, 22-27 May 2011, 1932-1935. https://doi.org/10.1109/ICASSP.2011.5946886

[20] Pongnumkul, S., Siripanpornchana, C. and Thajchayapong, S. (2017) Performance Analysis of Private Blockchain Platforms in Varying Workloads. 2017 26*th International Conference on Computer Communication and Networks* (*ICCCN*), Vancouver, BC, 31 July-3 August 2017, 1-6. https://doi.org/10.1109/ICCCN.2017.8038517

[21] Castro, M. and Liskov, B. (1999) Practical Byzantine Fault Tolerance. *Proceedings of the Third Symposium on Operating Systems Design and Implementation*, New Orleans, 22-25 February 1999, 173-186.

[22] Zaman, I. and He, M. (2021) A Multilayered Semi-Permissioned Blockchain Based Platform for Peer to Peer Energy Trading. 2021 *IEEE Green Technologies Conference* (*GreenTech*), Denver, CO, 07-09 April 2021, 279-285. https://doi.org/10.1109/GreenTech48523.2021.00052

[23] Poullikkas, A., Kourtis, G. and Hadjipaschalis, I. (2013) A Review of Net Metering Mechanism for Electricity Renewable Energy Sources. *International Journal of Energy and Environment* (*Print*), **4**, 975-1002.

[24] Dorri, A., Hill, A., Kanhere, S., *et al.* (2019) Peer-to-Peer Energytrade: A Distributed Private Energy Trading Platform. 2019 *IEEE International Conference on Blockchain and Cryptocurrency* (*ICBC*), Seoul, 14-17 May 2019, 61-64. https://doi.org/10.1109/BLOC.2019.8751268

[25] Hyperledger Composer. https://www.hyperledger.org/use/composer