Leveraging a Smart Grid Framework for Interoperability with a Distributed Ecosystem

Joan Ferré-Queralt*, Farzana Kabir†, David Megías†, Alexandre Viejo*, Jordi Castellà-Roca*

* Universitat Rovira i Virgili
Dept. d'Enginyeria Informàtica i Matemàtiques, CYBERCAT,
Av. Països Catalans 26, E-43007 Tarragona, Spain
{joan.ferreq, alexandre.viejo, jordi.castella}@urv.cat

† Universitat Oberta de Catalunya
Internet Interdisciplinary Institute (IN3), CYBERCAT,
Rambla del Poblenou, 154, 08018, Barcelona, Spain
{fkabir,dmegias}@uoc.edu

***Abstract*—The transition from traditional, centralized power grids to decentralized smart grids is essential for integrating renewable energy sources and enhancing the efficiency, reliability, and sustainability of electricity production and distribution. This paper presents a comprehensive evaluation framework for a decentralized smart grid demonstrator, addressing key challenges such as interoperability, security, privacy, and scalability. The proposed architecture to perform the evaluation comprises three layers: user interface nodes (prosumers and smart meters), distribution nodes, and a master node, facilitating bidirectional electricity trading and communication. We implement this testing framework using a network of Raspberry Pi devices and a personal computer, simulating real-world smart grid conditions. The experimental evaluation, conducted with a dataset of household electricity consumption and production, confirms the system's ability to simulate diverse user behaviors, maintain configurability, support various implementations, and ensure modularity and interoperability. Our results demonstrate the system's effectiveness in providing a robust platform for evaluating different smart grid implementations, focusing on user anonymity, data integrity, system reliability, computational cost, and electricity consumption.**

***Index Terms*—smart grid, testing environment, distributed, interoperability, privacy, anonymity, prosumers**

I. Introduction

The traditional power grid has been characterized by a unidirectional flow of electricity from generally centralized, non-renewable power plants to consumers, whose primary role was to consume electricity. Recently, the integration of renewable energy sources has significantly increased to meet growing demand without relying on non-renewable resources [1], [2]. In this sense, according to the International Renewable Energy Agency and the International Energy Agency [1], renewable energy sources will account for over 42% of global electricity generation, with the share of wind and solar PV doubling to 25% by 2028. This shift has paved the way for new and innovative energy solutions.

One of the most notable trends is the rise of Distributed Energy Resources (DERs) [3]. Unlike traditional power plants, DERs are installations that households can own, such as rooftop solar panels or small wind turbines. With these installations, end users can produce their own energy, use it immediately, or return excess energy to the grid. This has introduced the concept of the prosumer—users who both consume and produce electricity. This dual role is reshaping the traditional power grid, moving it towards obsolescence, and making way for a more dynamic and interactive system with a bidirectional flow [4].

The transformation from a centralized model to a distributed energy system has led to the development of the smart grid, which employs digital communication technologies to enhance the efficiency, reliability, and sustainability of electricity production and distribution [5], [6]. It is designed to integrate a variety of energy sources, including solar, wind, and other renewables, while facilitating real-time communication and control across the network [7]. A crucial aspect of this transformation is ensuring interoperability within the smart grid, enabling seamless communication between multiple entities of the system [8]–[10].

Implementing a decentralized smart grid involves multiple challenges. The incorporation of decentralized applications leveraging blockchain technology offers further enhancements in security, transparency, and autonomous operations in the smart grid [11]–[13]. However, scalability issues arise from managing massive data flow and network load from numerous devices. Security and privacy are significant concerns, requiring robust measures to protect the confidentiality, integrity, and authenticity of prosumer data [14].

To evaluate these challenges in a smart grid architecture, this paper proposes a testing environment for a distributed smart grid, focusing on security, privacy, and efficiency. The demonstrator aims to showcase various smart grid system implementations and address key concerns by evaluating user anonymity, data integrity, and system reliability while evaluating computational cost and electricity consumption associated with different consensus mechanisms and communication protocols.

II. State of the Art

In light of the above, this section reviews the current state of demonstrators for smart grids with a particular emphasis on privacy concerns, highlighting key projects, methodologies, and frameworks implemented to address these issues.

A comprehensive review of smart grid policies by Brown et al. [15] underscores the evolving landscape of regulatory

measures aimed at protecting consumer privacy while developing smart grids. The study identifies the dual challenge of achieving technological advancement and ensuring privacy protection, highlighting various demonstration projects that leverage existing smart grid infrastructure. However, these demonstrators have more focus on the electrical infrastructure aspect, with a broader view of the smart grid.

The Los Angeles Smart Grid Demonstration Project provides a critical analysis of security and privacy issues associated with smart grid software, particularly those deployed on cloud platforms. This project focuses on identifying vulnerabilities and proposing robust security frameworks to mitigate potential risks. The findings emphasize the need for comprehensive security measures to protect sensitive consumer data from cyber threats [16]. However, this project only addresses cloud-based smart grid implementations.

In [17], several projects of smart grid implementations are listed. However, most frameworks have only been developed to test specific implementations. Little research has been done to create a general framework for different privacy-preserving smart grid implementations.

The reviewed literature and demonstration projects highlight significant advancements in the development of technologies for smart grids. While progress has been made, ongoing research and development are essential to address the challenges in this field. The development of a general framework to test various smart grid implementations, focusing more on factors such as privacy, scalability, and computational cost, could be beneficial for the research sector.

III. Overview of the Architecture

In order to study the challenges inherent in smart grids, we propose a testing environment for smart grid implementations. This framework allows us to develop a demonstrator to evaluate the requirements of bidirectional electricity trading systems. The environment is a three-layer architecture that includes various actors and components, each playing a crucial role in the system's operation. These are called user interface nodes, distribution nodes, and master node, which can be described as follows:

1) *User Interface Nodes*: It comprises both prosumers and smart meters. Prosumers are the end users who can generate and consume electricity from renewable sources like wind turbines and solar panels. They support the smart grid's ability to balance supply and demand. Smart meters are IoT devices that measure how much electricity each prosumer produces and uses, giving a real-time record for controlling and monitoring energy use. These meters simulate prosumers by storing consumption and production data and using anonymization techniques before sending it to the distribution node.
2) *Distribution Nodes*: This middle layer acts as an intermediary between prosumers and the transmission layer of the grid. They maintain electrical infrastructure, ensure reliable electricity delivery, and aggregate data from multiple prosumers to coordinate electricity distribution. Prior to transmitting the data to the master node, these nodes compute assessment metrics, gather both real and anonymized data from user interface nodes, and apply additional anonymization algorithms if needed.
3) *Master Node*: This is the main node that regulates the entire operation, adjusts different parameters, adds or removes nodes, starts or stops the system as necessary, and performs comprehensive analysis based on data and evaluation metrics from distribution nodes. Additionally, it provides guidance on the particular smart grid deployment to the distribution node and user interface node.

Fig. 1 depicts the basic architecture.

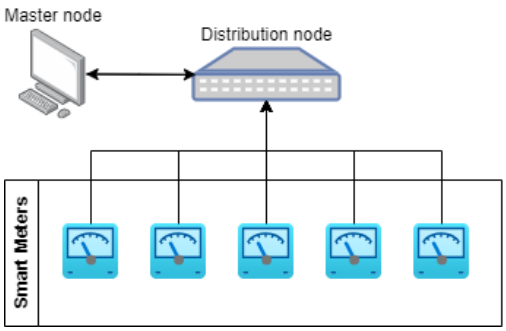


Fig. 1. System architecture

A. Operational Phases

The proposed simulation framework operates through four operational phases. These phases are outlined as follows:

1) *Setup*: This initial phase involves the preparation and configuration of all nodes within the system. Each node, including user interface nodes, distribution nodes, and the master node, is equipped with the necessary software and datasets to participate effectively in the simulation. Once configured, all nodes register with the master node and await for the simulation to start.
2) *Test Definition*: In this phase, the master node defines the parameters for the simulation, setting up the conditions under which the smart grid will operate. The master node also assigns smart meters to their respective distribution nodes.
3) *Energy Trading*: During this phase, a simulation of electricity production and consumption takes place. Prosumers generate and consume electricity, with smart meters recording and reporting these activities in real-time. The user interface nodes communicate that data to the distribution nodes.
4) *Data Gathering*: The final phase focuses on the comprehensive collection and analysis of data. Distribution nodes gather anonymized and raw data from the smart meters and transmit this information to the master node. The master node is then able to perform a thorough evaluation of the data, assessing key performance metrics such as energy efficiency, system reliability, and privacy preservation. This phase is useful to identify areas for improvement and to validate the effectiveness of the implemented smart grid solutions.

B. Requirements

Given all these considerations, we propose a simulation involving a group of users, where each user is represented by a smart meter. This setup aims to explore the practical implementation of a distributed smart grid, focusing on the

privacy and efficiency of distributed energy trading systems. Through this system, we enable the evaluation of key concerns such as user anonymity, data integrity, system reliability, computational cost, and electricity consumption.

The following requirements are identified for the proposed demonstrator to evaluate different smart grid implementations:

R1 *User simulation*: The system must support the simulation of different users (prosumers) reporting their electricity consumption and production. Each simulated user should have the capability to report data to the system, enabling the analysis of various scenarios and the performance of the smart grid under different conditions.

R2 *Configurable*: Parameters such as the size of the time slot, the simulation interval, and the number of users or groups of users should be able to be modified and tuned for each simulation.

R3 *Generic*: The system should be able to test different implementations of smart grid proposals, with different anonymization and aggregation techniques.

R4 *Modularity*: The system should have several components, being able to add or delete any of them. This should enable different simulation configurations.

R5 *Interoperability*: The system must ensure seamless communication and data exchange between different components.

R6 *Collection and metrics*: The demonstrator must support collection and analysis of data from smart meters. This data includes electricity consumption and production metrics. Thanks to the data provided, it should be able to perform analysis that may include anonymity, integrity, data utility, system scalability, and computational cost.

IV. Simulation Scheme

In this section, we describe the different steps that our demonstrator follows, which are: setup, test definition, electricity trading, and data gathering. Each of these steps are explained below.

A. Setup

In the setup phase, the necessary code and datasets are distributed to the respective nodes in the system.

1) Smart Meters: Each smart meter is given:

- *Code for Data Collection*: The smart meters are equipped with scripts to collect electricity consumption and production data.
- *Anonymization Code*: Scripts to anonymize the collected data to ensure privacy. This scripts are different for each implementation to be tested.
- *Datasets*: Historical data for electricity consumption and production.
- *Master address*: IP address of the master node.

2) Distribution Nodes: Each distribution node receives:

- *Management Code*: Scripts to manage the smart meters assigned to it.
- *Aggregation Code*: Scripts to aggregate data from multiple smart meters. These scripts are different for each implementation to be tested.
- *Master address*: IP address of the master node.

3) Master Node: The master node is provided with:

- *Initialization Code*: Scripts to initialize the simulation environment.
- *Registration Code*: Scripts to register smart meters and distribution nodes.
- *Management Code*: Scripts to manage the entire simulation process.

B. Test Definition

During the test definition phase, the master node initializes the simulation environment and registers the smart meters and distribution nodes. Detailed steps include:

1) Initialization: The master node starts the server and waits for connections from distribution nodes and smart meters.

Algorithm 1 Master Node Initialization

```
1:  procedure StartServer
2:      Initialize server socket
3:      Bind socket to host and port
4:      Listen for incoming connections
5:      while not exit do
6:          Accept connection
7:          Receive data
8:          if data type is 'register' then
9:              if role is 'distribution node' then
10:                 Register distribution node
11:             else if role is 'smart meter' then
12:                 Register smart meter
13:             end if
14:         else if data type is 'metrics' then
15:             Process metrics
16:         end if
17:         Send acknowledgment
18:     end while
19: end procedure
```

2) Registration: Smart meters and distribution nodes register with the master node, providing their respective addresses and ports.

- **Smart Meter Registration**: Each smart meter sends a registration message to the master node with its ID, host, and port.
- **Distribution Node Registration**: Each distribution node sends a registration message to the master node with its ID, host, and port.

3) Grouping: The master node groups distribution nodes with smart meters based on predefined criteria. Each distribution node is assigned a set of smart meters.

4) Simulation Parameters: The master node decides the timeslot length and the time interval for the simulation. These parameters dictate the granularity and duration of data collection.

- **Timeslot Length**: The duration of each data collection period.
- **Time Interval**: The total duration of the simulation.

C. Energy Trading

In the energy trading phase, smart meters begin to send their electricity consumption and production data to the assigned distribution nodes. The detailed process is as follows:

Algorithm 2 Grouping Devices

```
1: procedure GROUPDEVICES
2:     if no distribution nodes or smart meters then
3:         Log "No distribution nodes or smart meters to group." return
4:     end if
5:     Display grouping dialog to the master node
6:     Group smart meters to distribution nodes
7:     for each distribution node do
8:         Assign smart meters to the distribution node
9:     end for
10: end procedure
```

1) Data Request: Distribution nodes request data from their assigned smart meters, specifying the timeslot length and time interval.

Algorithm 3 Request Data from Smart Meters

```
1: procedure REQUESTDATA
2:     for each smart meter do
3:         Create request message with timeslot length and time interval
4:         Send request to smart meter
5:     end for
6: end procedure
```

2) Data Transmission: Smart meters respond to the requests by sending the actual and anonymized data for consumption and production. This data is sent at regular intervals based on the timeslot length.

Algorithm 4 Send Data to Distribution Node

```
1: procedure SENDDATA
2:     while within simulation time interval do
3:         Collect consumption and production data
4:         Anonymize data
5:         Collect computational time of anonymizing data.
6:         Create data message
7:         Send data to distribution node
8:         Wait for next timeslot
9:     end while
10:    Send completion message to distribution node
11: end procedure
```

3) Data Aggregation: Distribution nodes collect data from all their assigned smart meters. Once all smart meters have transmitted their data, the distribution nodes aggregate the data to prepare it for the master node.

Algorithm 5 Aggregate Data

```
1: procedure AGGREGATEDATA
2:     Initialize total consumption and production
3:     for each smart meter's data do
4:         Add consumption and production to totals
5:     end for
6:     Create aggregated data message
7:     Send aggregated data to master node
8: end procedure
```

V. Data Gathering

The final phase involves gathering all the collected data and performing analysis. The master node takes the following steps:

A. Data Collection

Distribution nodes send the aggregated data to the master node.

Algorithm 6 Collect Data from Distribution Nodes

```
1: procedure COLLECTDATA
2:     for each distribution node do
3:         Receive aggregated data
4:     end for
5:     Store received data
6: end procedure
```

1) Data Processing: The master node processes the received data, enabling it to calculate metrics such as total consumption, production, and utility loss.

Algorithm 7 Process Received Data

```
1: procedure PROCESSDATA
2:     Initialize metrics storage
3:     for each received data do
4:         Calculate total consumption and production
5:         Calculate utility loss
6:         Store metrics
7:     end for
8: end procedure
```

VI. Experimental Evaluation

In this section, we conduct an experimental evaluation of our proposal with a real implementation of the steps detailed in the previous sections.

A. Test Environment

Our implementation employs a set of Raspberry Pi devices to act as distribution nodes and user interface nodes. This choice is motivated by the aim to closely simulate the computational capacity and electricity consumption of IoT devices typically used as smart meters.

For the master node, we used a personal computer. This decision was made to ensure easy communication with all other nodes and to facilitate the visualization and verification of the data sent by the distribution nodes and smart meters. The master node's higher computational power and enhanced connectivity options provide a stable platform for managing simulation.

To enable communication between all components, a Local Area Network (LAN) was established. A predefined set of IP addresses and ports was assigned to each component to enable the message exchange process. Communication was facilitated via WiFi.

All scripts were developed using the Python programming language. Python was selected due to its extensive libraries and tools for data analysis, making it an ideal choice for future data processing and analysis tasks. Key libraries used include:

- **Socket Programming**: For managing network communications.
- **Pandas**: For handling and processing data.
- **Matplotlib**: For data visualization.

To facilitate the configuration of different tests and the visualization of data, a Graphical User Interface (GUI) was implemented on the master node using the Tkinter library, as seen in Fig. 2. The GUI allows users to:

- Configure simulation parameters such as timeslot length and time interval.
- Monitor the registration and status of distribution nodes and smart meters.
- Visualize simulation data transmission.

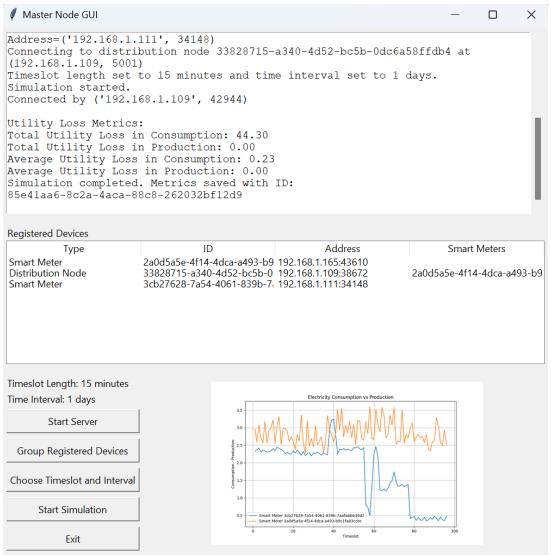


Fig. 2. Screenshot of the GUI implemented on the master node using Tkinter.

All the data (real, anonymized, and aggregated) is stored in a JSON file in the master node for future analysis.

B. Dataset

To execute our experiment, we utilized the dataset sourced from [18] which provided comprehensive data on energy consumption and photovoltaic (PV) generation for a total of 51 buildings (50 residential households and a public building). With a 15-minute time interval, the fine-grained data enabled precise simulation of real-life energy usage patterns from different types of prosumers.

C. Requirements Evaluation

In this subsection, we evaluate our implementation against the requirements outlined in Section VI-C. The evaluation focuses on key aspects such as user simulation, configurability, generality, modularity, interoperability, and data collection and metrics.

R1 *User Simulation*: The system effectively supports the simulation of different users (prosumers) reporting their electricity consumption and production. Each simulated user, represented by a smart meter, reports data to the system. The implementation demonstrated the ability to simulate diverse user behaviors and their impact on the grid.

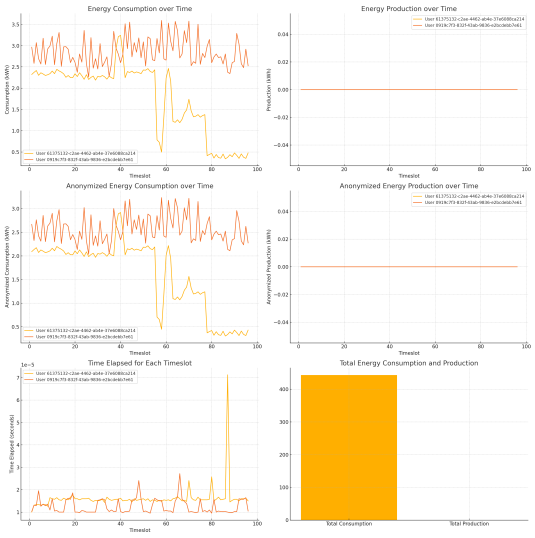


Fig. 3. JSON output analysis

R2 *Configurability*: The system allows for the modification and tuning of parameters such as the size of the time slot, the simulation interval, and the number of users or groups of users for each simulation. The GUI on the master node provides an intuitive interface for configuring these parameters, ensuring that different simulation setups can be easily tested. This flexibility was confirmed through multiple simulation runs with varying configurations.

R3 *Generality*: The system is designed to test different implementations of smart grid proposals, with various anonymization and aggregation techniques. The modular nature of the anonymization and aggregation code allows for the easy integration and testing of different methods.

R4 *Modularity*: The architecture supports the addition or removal of components, enabling different simulation configurations. This was verified by adding and removing smart meters and distribution nodes.

R5 *Interoperability*: The use of standardized communication protocols and the handling of data transmissions between smart meters, distribution nodes, and the master node were validated through successful data exchanges during the simulations. The system maintained interoperability across all components.

1) Data collected: To thoroughly evaluate the system's data collection and metrics capabilities, we conducted detailed analyses using the collected JSON data. The analysis focused on various critical aspects, including anonymity, data utility, system scalability, and computational cost. Below are the key findings from these evaluations also depicted in Fig. 3

By comparing the anonymized data with the real data, we successfully analyzed the effectiveness of the anonymization techniques employed. We were also capable of measuring data utility by examining how closely the anonymized data matched the real data in terms of aggregate metrics such as total energy consumption and production.

By analyzing the time elapsed for each transaction and focusing on the system's ability to handle an increasing number of smart meters and distribution nodes, the user can evaluate if the system maintains acceptable performance levels

with a moderate increase in computational load, suggesting good scalability and low computational cost.

VII. Conclusions

This paper presents a comprehensive framework for a distributed smart grid demonstrator, which gathers data to evaluate key challenges such as interoperability, security, privacy, and scalability. The proposed architecture, consisting of user interface nodes, distribution nodes, and a master node, provides a robust platform for evaluating various smart grid implementations.

The experimental evaluation, conducted using a real dataset and a network of Raspberry Pi devices, confirmed that the system meets the outlined requirements. The results demonstrated the system's effectiveness in simulating various user behaviors, maintaining configurability, supporting multiple implementations, and ensuring modularity and interoperability.

A. Future work

Future work will focus on enhancing the system's scalability and robustness. Potential areas for improvement include:

- **Real-Time Data Processing**: Implementing real-time data processing capabilities to enable immediate decision-making and grid management.
- **Extended Metrics Analysis**: Incorporating additional metrics for a more comprehensive evaluation of smart grid performance, including environmental impact and economic efficiency.

Acknowledgments

This work was partially supported by the Spanish Ministry under Grant PID2021-125962OB-C32 "SECURING/DATA" funded by MCIN/AEI/10.13039/501100011033/ FEDER, UE, by the grant 2021SGR 00115 from the Government of Catalonia, and by INCIBE and the European Union NextGenerationEU/Plan de Recuperación, Transformación y Resiliencia through the projects "HERMES" and INCIBE-URV cybersecurity chair.

Additionally, this research is supported by the Detection of fake newS on SocIal MedIa pLAtfoRms (DISSIMILAR) from the EIG CONCERT-Japan (PCI2020-120689-2, Government of Spain), the "SECURING" project (PID2021-125962OB-C31) funded by the Ministerio de Ciencia e Innovación, la Agencia Estatal de Investigación and the European Regional Development Fund (ERDF), as well as the ARTEMISA International Chair of Cybersecurity (C057/23) and the DANGER Strategic Project of Cybersecurity (C062/23), both funded by the Spanish National Institute of Cybersecurity through the European Union – NextGenerationEU and the Recovery, Transformation and Resilience Plan.

References

[1] IEA. (2023) World energy outlook 2023, iea paris. [Online]. Available: https://www.iea.org/reports/world-energy-outlook-2023

[2] C. P. Ohanu, S. A. Rufai, and U. C. Oluchi, "A comprehensive review of recent developments in smart grid through renewable energy resources integration," *Heliyon*, 2024.

[3] D. M. L. González and J. G. Rendon, "Opportunities and challenges of mainstreaming distributed energy resources towards the transition to more efficient and resilient energy markets," *Renewable and Sustainable Energy Reviews*, vol. 157, p. 112018, 2022.

[4] J. Hu, Y. Li, and H. Zhou, "Energy management strategy for a society of prosumers under the iot environment considering the network constraints," *IEEE Access*, vol. 7, pp. 57 760–57 768, 2019.

[5] N. Hatziargyriou, "Frontiers in smart grids," *Frontiers in Smart Grids*, vol. 1, p. 986178, 2022.

[6] M. Khalid, "Smart grids and renewable energy systems: Perspectives and grid integration challenges," *Energy Strategy Reviews*, vol. 51, p. 101299, 2024.

[7] T. Kataray, B. Nitesh, B. Yarram, S. Sinha, E. Cuce, S. Shaik, P. Vigneshwaran, and A. Roy, "Integration of smart grid with renewable energy sources: Opportunities and challenges–a comprehensive review," *Sustainable Energy Technologies and Assessments*, vol. 58, p. 103363, 2023.

[8] N. Rosetto and V. Reif, "Digitalization of the electricity infrastructure: a key enabler for the decarbonization and decentralization of the power sector," in *A Modern Guide to the Digitalization of Infrastructure*. Edward Elgar Publishing, 2021, pp. 217–265.

[9] O. M. Butt, M. Zulqarnain, and T. M. Butt, "Recent advancement in smart grid technology: Future prospects in the electrical power network," *Ain Shams Engineering Journal*, vol. 12, no. 1, pp. 687–695, 2021.

[10] A. Lekbich, A. Belfqih, C. Zedak, J. Boukherouaa, and F. Elmariami, "Implementation of a decentralized real-time management system for electrical distribution networks using the internet of things in smart grids," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 3, pp. 1142–1153, 2021.

[11] M. Waseem, M. Adnan Khan, A. Goudarzi, S. Fahad, I. A. Sajjad, and P. Siano, "Incorporation of blockchain technology for different smart grid applications: Architecture, prospects, and challenges," *Energies*, vol. 16, no. 2, p. 820, 2023.

[12] V. Bolgouras, T. Ioannidis, I. Politis, A. Zarras, and C. Xenakis, "Retina: Distributed and secure trust management for smart grid applications and energy trading," *Sustainable Energy, Grids and Networks*, vol. 38, p. 101274, 2024.

[13] S. Anjimoon, R. Chandrashekar, N. Singh, A. Parmar, N. Sharma, and Q. Mohammad, "Secure and sustainable energy distribution through blockchain technology in smart grids," in *E3S Web of Conferences*, vol. 505. EDP Sciences, 2024, p. 02002.

[14] A. Upreti, J. Cardell, and D. Thiebaut, "Data privacy in the smart grid: a decentralized approach," 2019.

[15] A. e. a. Brown, "Smart grid governance: An international review of evolving policy," *Our Energy Policy*, 2021. [Online]. Available: https://www.ourenergypolicy.org/wp-content/uploads/2021/01/Brown_Smart-Grid-Governance.pdf

[16] L. A. S. G. D. Project, "An analysis of security and privacy issues in smart grid software," 2016. [Online]. Available: https://www.osti.gov/servlets/purl/1332299

[17] M. B. Mollah, J. Zhao, D. Niyato, K.-Y. Lam, X. Zhang, A. M. Ghias, L. H. Koh, and L. Yang, "Blockchain for future smart grid: A comprehensive survey," *IEEE Internet of Things Journal*, vol. 8, no. 1, pp. 18–43, 2020.

[18] C. Goncalves, R. Barreto, P. Faria, L. Gomes, and Z. Vale, "Dataset of an energy community's generation and consumption with appliance allocation," *Data in Brief*, vol. 45, p. 108590, 2022.