electronics

MDPI

Article

Advancing User Privacy in Virtual Power Plants: A Novel Zero-Knowledge Proof-Based Distributed Attribute Encryption Approach

Ruxia Yang [1,2,*], Hongchao Gao [3], Fangyuan Si [3] and Jun Wang [4]

[1] State Grid Smart Grid Research Institute Co., Ltd., Nanjing 210003, China
[2] State Grid Laboratory of Power Cyber-Security Protection and Monitoring Technology, Nanjing 210003, China
[3] State Key Laboratory of Power Systems, Department of Electrical Engineering, Tsinghua University, Beijing 100084, China; hcgaoth@tsinghua.edu.cn (H.G.); sifangyuan@mail.tsinghua.edu.cn (F.S.)
[4] State Grid Shanghai Municipal Electric Power Company, Shanghai 200122, China; wangjun@sh.sgcc.com.cn
* Correspondence: yangruxia@geiri.sgcc.com.cn

Abstract: In virtual power plants, diverse business scenarios involving user data, such as queries, transactions, and sharing, pose significant privacy risks. Traditional attribute-based encryption (ABE) methods, while supporting fine-grained access, fall short of fully protecting user privacy as they require attribute input, leading to potential data leaks. Addressing these limitations, our research introduces a novel privacy protection scheme using zero-knowledge proof and distributed attribute-based encryption (DABE). This method innovatively employs Merkel trees for aggregating user attributes and constructing commitments for zero-knowledge proof verification, ensuring that user attributes and access policies remain confidential. Our solution not only enhances privacy but also fortifies security against man-in-the-middle and replay attacks, offering attribute indistinguishability and tamper resistance. A comparative performance analysis demonstrates that our approach outperforms existing methods in efficiency, reducing time, cost, and space requirements. These advancements mark a significant step forward in ensuring robust user privacy and data security in virtual power plants.

Keywords: zero-knowledge proof; attribute hiding; virtual power plant; privacy protection; attribute-based encryption

Citation: Yang, R.; Gao, H.; Si, F.; Wang, J. Advancing User Privacy in Virtual Power Plants: A Novel Zero-Knowledge Proof-Based Distributed Attribute Encryption Approach. *Electronics* **2024**, *13*, 1283. https://doi.org/10.3390/electronics13071283

Academic Editor: Zbigniew Kotulski

Received: 5 February 2024
Revised: 11 March 2024
Accepted: 13 March 2024
Published: 29 March 2024

1. Introduction

In pursuit of carbon neutrality, China is undergoing a significant energy transformation, with a focus on integrating renewable energy sources within a robust smart grid framework [1,2]. This transformation involves the intricate coordination of generation, transmission, load, and storage systems, transitioning to both centralized and distributed generation models [1,3,4]. The envisioned power system aims to be sustainable, secure, adaptable, and efficient, heavily relying on new, cleaner energy sources [3,5–7].

Global electricity demands and environmental concerns highlight the inadequacies of traditional power generation. Renewable sources like wind and solar power have gained traction as sustainable alternatives. Yet, integrating these intermittent and distributed sources into the grid presents substantial challenges, notably in maintaining stable outputs and effective grid integration. Virtual power plants (VPPs) have emerged as a solution, orchestrating a myriad of distributed energy resources for improved renewable energy integration. This approach resonates with the comprehensive energy model encompassing generation, grid, load, and storage. However, existing VPP models are predominantly centralized in their management of information and transactions, creating vulnerabilities such as potential data manipulation and privacy breaches. Our research aims to tackle these issues through innovations in the realms of data security and privacy within VPP systems by leveraging advanced encryption and decentralized management methodologies.

As illustrated in Table 1, numerous domestic and international entities are actively investigating user data security within virtual power plants. In the United States, General Electric (GE) is pioneering the integration of blockchain technology with virtual power plants. This involves harnessing blockchain for the amalgamation of distributed energy generation and executing automated transactions via smart contracts, with the aim of achieving an equilibrium between supply and demand in the power grid. Another endeavor, SolarChange, is focused on developing a blockchain-based platform for energy interaction, targeting the energy balance challenge in virtual power plants and enhancing energy efficiency. In Japan, Kyocera is experimenting with the synergy of clean energy and energy storage systems in virtual power plants, utilizing blockchain as the foundational technology. Within China, a strategic partnership between Shanghai University of Electric Power and Shanghai Electric Power Company has been forged to expedite collaborative research in energy blockchain, including applications in virtual power plants. Additionally, the State Grid Corporation of China is leveraging its proprietary blockchain platform to streamline power trading, particularly in virtual power plant contexts.

Table 1. Application of blockchain technology in VPP at domestic and abroad.

Nation	Example	Technology Platform	Main Application Scenarios
America	TransActive Grid blockchain energy project	Ethereum blockchain	Virtual power plant distributed trading system and P2P trading
Australia	Ecochain	Ethereum blockchain	P2P trading system for PV surplus power
China	Energy blockchain lab	Hyperledger Fabric	Compressing costs in the virtual power plant trading and clearing process
Japan	Virtual power plant for solar and storage systems	LO3 Energy proprietary platform	Shared generation to reduce the burden on the grid
Spain	Blockchain tracking of the electricity distribution supply chain	TrustOS blockchain platform	Markets for renewable energy certification schemes
Estonia	Virtual power plant project based on power matchmaker	WePower platform	Promoting P2P interactions for clean energy
South Korea	Virtual power plant based on citizen sharing blockchain	Ethereum blockchain	Reduction of human error in record keeping

Despite these advancements, employing blockchain for user privacy protection in virtual power plants introduces several new challenges. Predominantly, existing methodologies facilitate only one-to-one data authorization and sharing. This necessitates the disclosure of one party's identity to the other with each data-sharing instance, potentially compromising both user privacy and system efficiency.

In response, researchers have explored the use of attribute-based encryption (ABE) to streamline data sharing and access between users and devices, as evidenced in the literature [8,9]. Hur and Noh [10] proposed a revocable attribute-based encryption system, utilizing a proxy for permission revocation, with the user list being publicly accessible to the proxy. Li et al. [11] applied scalable ABE in cloud computing for sharing personal health records. Khalil et al. [12] introduced a multi-authority ABE-based access control method,

enabling IoT devices to interact with IoT gateways. While these solutions utilize ABE for data access control, they inadequately address the prevention of privacy breaches during data sharing. The concept of anonymous ABE [13] has been proposed to mitigate attribute privacy leakage. However, this approach only secures access policies and leaves user attributes exposed, thereby still posing a risk of privacy infringement.

This paper endeavors to tackle the challenge of user privacy protection in virtual power plants. By thoroughly analyzing user behavioral patterns and identifying diverse business scenarios, we propose a novel privacy protection scheme founded on zero-knowledge proofs. The primary contributions of this work are delineated as follows.

1. We introduce an innovative data protection strategy named attribute-hiding zero-knowledge proof (AH-ZKP). This approach effectively conceals user identities and attributes during authorization and verification processes, enhancing privacy and security.
2. Our model innovatively integrates distributed attribute-based encryption (DABE) with multiple attribute management nodes. This design achieves decentralized attribute management, mitigating risks associated with single-point failures and collusion attacks common in centralized systems. We also introduce a decentralized data sharing scheme that preserves the confidentiality of user attribute privacy and ciphertext access policies while also facilitating auditing capabilities.
3. The effectiveness of our scheme is demonstrated through its ability to maintain the confidentiality of data, attributes, and access policies. It addresses key privacy concerns in virtual power plants, ensuring secure and private data transactions.

The rest of this paper is organized as follows. Section 1 provides an overview of virtual power plants and zero-knowledge proofs. Section 2 establishes the system model of the proposed scheme. Section 3 analyzes the security of the scheme, and Section 4 conducts experimental simulations. Finally, Section 5 provides a brief summary.

2. Research Background

2.1. Virtual Power Plant

The virtual power plant (VPP), a concept dating back over two decades, represents a significant shift in energy management and distribution. Originating in Europe, with countries like Germany, the United Kingdom, France, and the Netherlands leading the way, VPPs have been the focus of numerous mature demonstration projects. These projects primarily centered on integrating distributed energy into the grid reliably and establishing robust business models within the electricity market [7,14–16]. In China, the VPP concept has garnered increasing attention in recent years, spurred by the evolution of the nation's power systems and market policies, leading to substantive demonstration practices [1,3,16,17].

A virtual power plant, while lacking the physical infrastructure typical of traditional power plants, performs similar functions. It not only generates electricity but also actively participates in energy markets, contributing to grid stability through peak shaving and frequency regulation [18–20]. At its core, a VPP employs advanced information technologies to integrate diverse energy resources, including generation, consumption, and storage. This integration is achieved through collaborative efforts with external centralized control systems and management platforms, facilitating coordinated control, optimization, and enabling comprehensive data analysis for strategic operational adjustments [20]. VPPs actively engage in energy trading with various market players, responding dynamically to fluctuations in market demands. The operational model of a VPP can be likened to a 'black box' that simulates the functionalities of a physical power plant. It can act as a 'positive power plant' by supplying power to the grid or as a 'negative power plant' by absorbing excess power. This 'black box' encompasses distributed energy sources such as renewable energy, diverse energy storage facilities, electric vehicles, controllable loads, and more [1,3]. However, the mere aggregation of these resources is not sufficient for effective energy management. A VPP necessitates a comprehensive set of technologies and systems for the intelligent aggregation and management of these resources.

The essence of a virtual power plant lies in its innovative electricity management approach. It aggregates a multitude of distributed energy devices, including solar panels, wind turbines, and energy storage units, through advanced interconnectivity technologies, creating a virtual energy generation system. This system employs state-of-the-art intelligent control technologies for the efficient dispatch and management of distributed energy resources. Such management optimizes energy generation and usage, enhances the reliability and stability of the power grid, and reduces operational and maintenance costs. Virtual power plants typically utilize cloud-based platforms for data aggregation and sophisticated analysis, thereby elevating the intelligence and efficiency of the power system. The continuous evolution of technologies like 5G, artificial intelligence, and cloud computing is expanding the application and potential of virtual power plants, promising a future of widespread global adoption and advancement in the energy sector.

2.2. Zero-Knowledge Proofs

Zero-knowledge proof (ZKP) is a cryptographic protocol enabling one party, the prover, to demonstrate the veracity of a proposition to another, the verifier, without divulging any additional information. This ensures the prover's privacy, safeguarding sensitive data from being inferred by the verifier during the proof process.

ZKP is characterized by three essential properties:

1. **Completeness**: Given correct evidence or a 'witness', the prover can convincingly demonstrate the truthfulness of the assertion to the verifier with high probability.
2. **Soundness**: A deceptive prover, lacking the correct witness, finds it implausibly challenging to convince the verifier of a false proposition.
3. **Zero-knowledge**: The verifier gains no additional information, apart from the proposition's validity, from the proof procedure.

ZKPs are classified into two types: interactive and non-interactive. Interactive ZKPs entail several rounds of communication between the prover and verifier, with the verifier posing random challenges and the prover responding accordingly. Conversely, non-interactive ZKPs require just a single communication round. Here, the prover sends a comprehensive message to the verifier, which diminishes communication complexity and enhances efficiency. The zero-knowledge succinct non-interactive argument of knowledge (zk-SNARK) is the non-interactive ZKP variant employed in this study.

We focus on the Groth16 algorithm [21] for zk-SNARK implementation, constructing pairing-based non-interactive zero-knowledge (NIZK) proofs within a quadratic arithmetic program (QAP) framework. Initially, we define the relation-generating element R, expressed as follows:

$$R = (p, G_1, G_2, G_T, e, l, \{u_i(X), v_i(X), w_i(X)\}_{i=0}^{m}, t(X)) \tag{1}$$

Here, $|p| = \lambda$, where p represents the size of the finite field, and λ is the security parameter. The degrees of u, v, and w are $n-1$, and of t is n. The relation R defines the domain Z_p, the statements $(a_1, \ldots, a_l) \in Z_p^l$, and the witnesses $(a_{l+1}, \ldots, a_m) \in Z_p^{m-l}$. These components ensure that, given $a_0 = 1$, for the $n-2$ degree quotient polynomial $h(X)$, the following equation holds:

$$\sum_{i=0}^{m} a_i u_i(X) \sum_{i=0}^{m} a_i v_i(X) = \sum_{i=0}^{m} a_i w_i(X) + h(X)t(X) \tag{2}$$

Building upon the zk-SNARK implementation via the Groth16 algorithm, we delineate the following pairing-based non-interactive zero-knowledge (NIZK) argument protocol:

Setup $(R) \rightarrow (\sigma, \tau)$:
Pick $\alpha, \beta, \gamma, \delta, x \leftarrow Z_p^*$.
Set $\tau = (\alpha, \beta, \gamma, \delta, x)$

$$\sigma = \begin{pmatrix} G^{\alpha}, G^{\beta}, H^{\beta}, H^{\gamma}, G^{\delta}, H^{\delta}, \{G^{x^i}\}_{i=0}^{n-1}, \{H^{x^i}\}_{i=0}^{n-1} \\ \{G^{\frac{\beta u_i(x)+\alpha v_i(x)+w_i(x)}{\gamma}}\}_{i=0}^{l}, \{G^{\frac{\beta u_i(x)+\alpha v_i(x)+w_i(x)}{\delta}}\}_{i=l+1}^{m}, \{G^{\frac{x^i t(x)}{\delta}}\}_{i=0}^{n-2} \end{pmatrix} \tag{3}$$

Prove $(R, \sigma, a_1, \ldots, a_m) \rightarrow \pi$:
Pick $r, s \leftarrow Z_p$.
Compute $\pi = (A, B, C)$ where
$A = G^{\alpha+\sum_{i=0}^{m} a_i u_i(x)+r\delta}$, $B = H^{\beta+\sum_{i=0}^{m} a_i v_i(x)+s\delta}$

$$C = G^{\frac{\sum_{i=l+1}^{m} a_i\left(\beta u_i(x)+\alpha v_i(x)+w_i(x)\right)+h(x)t(x)}{\delta}+s\left(\alpha+\sum_{i=0}^{m} a_i u_i(x)\right)+\gamma\left(\beta+\sum_{i=0}^{m} a_i v_i(x)\right)-rs\delta}$$

Verify $(R, \sigma, a_1, \ldots, a_m) \rightarrow b$:
Compute a quadratic multi-variate polynomial, such that $\pi = (A, B, C)$ corresponds to the test.

$$e(A, B) = e(G^{\alpha}, H^{\beta}) e\left(G^{\frac{\sum_{i=l+1}^{m} a_i(\beta u_i(x)+\alpha v_i(x)+w_i(x))}{\gamma}}, H^{\gamma}\right) e(C, H^{\delta}) \tag{4}$$

Accept the proof if the test passes.

This protocol exemplifies a non-interactive zero-knowledge proof with complete and perfect zero-knowledge properties. It is among the fastest and most compact zk-SNARKs available, necessitating only three proof elements and featuring a straightforward verification equation. This ensures robust integrity and reliability under polynomial computation capabilities, making it an optimal choice.

3. User Attribute Hiding Model Based on Zero-Knowledge Proof

3.1. Problem Analysis

Virtual power plants (VPPs) epitomize the integration of diverse energy sources, encompassing conventional, renewable, and micro-energy systems. They play a pivotal role in balancing power generation with load demands by managing resources like energy storage, electric vehicles, and controllable loads. Moreover, VPPs engage in trading surplus energy, either to other VPPs or the public grid, optimizing electricity sale revenue and internal power balance while curbing operational costs. Consequently, safeguarding the privacy of VPP users becomes paramount.

Existing studies on data sharing in VPPs predominantly target the protection of data content. Attribute-based encryption (ABE) is employed for this purpose, offering several advantages:

1. ABE reduces encryption overhead as it does not necessitate individual encryption for each recipient, thereby streamlining the process.
2. It heightens data security. Even in scenarios of untrustworthy or compromised data servers, ABE ensures information confidentiality.
3. ABE supports fine-grained access control, enabling the creation of bespoke access structures tailored to specific scenarios and requirements.
4. The design of ABE thwarts collusion attacks by preventing the simultaneous use of different users' private keys.

However, the predominant focus of ABE methodologies on data content protection reveals inherent limitations. While encrypted data remain publicly accessible, malicious entities can exploit this visibility. Through record analysis, they can intercept the access policies of specific files or ascertain the attributes of particular users. Employing strategies like correlation and inference attacks or clustering analysis, these entities can potentially deduce sensitive user information or identities. Thus, it is imperative to extend protection beyond data security and traceability. Concealing user attributes, authorization relationships, and access policies is crucial to ensuring the sanctity of both data content and member privacy.

Table 2 in this paper elucidates the notations used in our discussion, aiding in a clearer understanding of these complex concepts.

This section combines the privacy attributes of virtual power plant users with different scenarios and utilizes distributed attribute-based encryption technology to propose an attribute hiding scheme.

Table 2. Symbol explanation.

Symbol	Meaning of Symbols
$RawData$	Raw data
$EncData$	Coded text
λ, G, P	Global parameters required by DABE
K_{prov}, K_{veri}	Proof keys and verification keys in zero-knowledge proofs
PK, SK	Public–private key pair
$addr$	User address
$AttrList$	List of properties used in DABE and zero-knowledge proofs
$AttrRoot$	Root node of the attribute Merkle tree
$attr_i$	User attributes
$K_{i,uid}$	DABE key corresponding to $attr_i$
r_{random}	Random number
PSF	Pseudo-random sorting function
CRH	Collision-resistant hash function
$COMM$	Non-interactive commitments
AH-ZKP	Zero-knowledge proofs of property hiding
$COMM_{attr}$	Non-interactive commitments generated by user attributes and random numbers
$COMM'_{attr}$	Non-interactive commitments generated from user attributes and addresses
$Cert_{veri}$	Certificates required to download data
$informariondownloader$	Information for data visitors

3.2. User Privacy Attributes

In the realm of virtual power plants (VPPs), user privacy data are multifaceted, informed by the data analysis of the VPP business system. These data encompass the following:

- Personal information: This includes the user's name, address, telephone number, email, and other identifying details.
- Power consumption data: Critical data such as real-time power usage, load profiles, and consumption timings.
- Electrical equipment data: Information on the user's electrical devices, including type, brand, model, and age.
- Behavioral data: Patterns and habits in electricity usage, like usage times and modes.

The sensitivity of these data points is evident; unauthorized access or breaches can lead to significant property losses or even physical safety risks. Hence, VPPs must implement rigorous data protection strategies to safeguard the confidentiality and integrity of user data.

3.3. Scene Analysis

The protection of user privacy data in VPPs spans multiple scenarios, each with its unique challenges:

- **Exchange and sharing scenarios**: Company data sharing, which is vital for analytics, auditing, or training, entails privacy risks. The de-identification of data is crucial to maintaining privacy integrity during internal exchanges.
- **Development and testing scenarios**: R&D and testing phases necessitate large volumes of raw data. To prevent privacy breaches, these data must undergo de-identification before being used in these environments.
- **External release scenarios**: The external dissemination of data, a key facet of the VPP data business, mandates stringent privacy safeguards. This includes the thorough analysis and de-identification of user data, ensuring no privacy leaks occur during external interactions or transactions.

This enhanced section underlines the criticality of protecting user privacy in diverse operational scenarios of VPPs, highlighting the need for specialized data handling practices to mitigate privacy risks.

3.4. System Model

This section intricately details a data sharing model within a virtual power plant ecosystem, leveraging an advanced zero-knowledge proof-based attribute hiding and policy protection mechanism. This sophisticated model interweaves cryptographic techniques and network entities, forming a bulwark for privacy and security in data sharing.

Our model addresses the three main privacy issues mentioned in the previous section. The system's architecture, illustrated in Figure 1, consists of seven integral entities at a logical level: registration and authentication node (RAN), multiple attribute management node (AMN), data uploader, data downloader, data center (DC), virtual power plant console (VPP), and review node (AN). Each entity is meticulously designed to fulfill critical functions in the data handling process, ensuring secure and private data management.

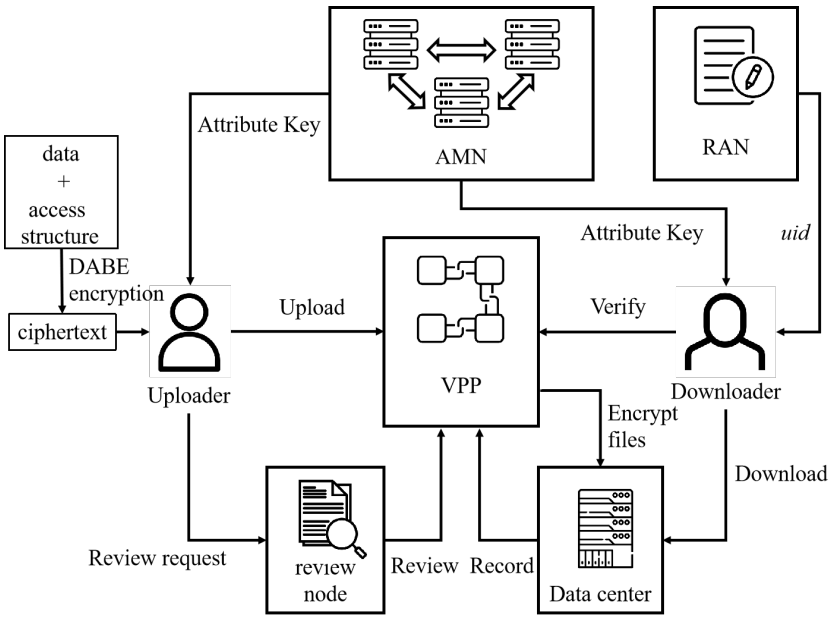

Figure 1. A user attribute privacy protection model based on zero-knowledge proof.

1. **Registration and authentication node (RAN):** This node is instrumental during the system's initialization. It generates a unique identity *uid* for each user, which is linked to a randomly generated value. This identity is crucial for user verification across

the system, and its one-time activation mechanism, coupled with its autonomy from central authorities, augments the system's security and reliability.

2. **Multiple attribute management nodes (AMNs):** These decentralized nodes are responsible for specific attribute domains. They generate private keys (SKs) for each attribute, binding these keys to the user's *uid*. This decentralized structure reduces the risks associated with centralized control, such as single-point failures and potential security breaches.
3. **Data uploader:** The uploader employs the distributed attribute-based encryption (DABE) algorithm for encrypting data. DABE is chosen for its efficiency in managing multiple attributes and its capability of enforcing granular access control policies. The uploader encrypts the data and then transmits the ciphertext to the smart contract, which in turn provides the data storage address.
4. **Data downloader:** Utilizing the attribute hiding zero-knowledge proofs (AH-ZKPs), this entity demonstrates compliance with access control policies without revealing specific attribute values. AH-ZKP is pivotal in preserving privacy while facilitating access to data storage addresses. The downloader decrypts the ciphertext using attribute keys, ensuring secure data retrieval.
5. **Data center (DC):** The DC is a nexus between the nodes and data storage. It receives encrypted data from the nodes, stores them, and returns the data storage address. For data retrieval, it verifies download requests through smart contracts and releases the encrypted data upon successful validation, adding an essential security layer.
6. **Virtual power plant console (VPP):** The VPP serves as an intermediary in data transmission. It handles encrypted files from uploaders, verifies attribute commitments against access control policies, and maintains access control lists (ACLs). These ACLs are crucial for tracking commitments and managing data storage, playing a significant role in the system's data governance.
7. **Review node (AN):** The AN is the system's audit and compliance overseer. In instances of data disputes or irregularities, the AN intervenes to provide an audit trail, ensuring transparency and accountability in data operations within the virtual power plant.

The elaborated model, with its emphasis on decentralization, zero-knowledge proofs, and attribute-based encryption, represents a state-of-the-art approach to data security and privacy in the context of virtual power plants. It addresses contemporary challenges in user privacy and data security with a comprehensive, innovative framework.

To safeguard the encrypted data stored in data centers, it is imperative to verify the data visitors' permissions against the access control list (ACL) before granting them the file's storage address. This section introduces a novel data access control strategy, the attribute-hiding zero-knowledge proof (AH-ZKP). AH-ZKP synergizes zero-knowledge proof with attribute-based encryption to enable attribute-hidden access control. This strategy surpasses the conventional attribute-based encryption methods by adding an extra layer of privacy, thereby validating user privileges without compromising the privacy of user attributes and access policies.

The implementation of AH-ZKP faces three primary challenges. Firstly, it requires a seamless integration of zero-knowledge proofs with user attributes to ensure robust privacy protection. Secondly, it involves utilizing zero-knowledge proofs for validating access policies, which are composed of user attributes, in a manner that circumvents potential security threats like re-entry and man-in-the-middle attacks. Lastly, deploying AH-ZKP within a virtual power plant setting demands careful consideration to maintain the trustworthiness of the validation results.

To address the outlined challenges, we introduce an attribute-based Merkle tree structure designed for the efficient aggregation of user attributes. Utilizing the Merkle root, we construct commitments within the zk-SNARK framework, enabling zero-knowledge proof verification. This approach ensures that verifiers ascertain compliance with access policy requirements without gaining insight into specific user attributes. Moreover, by

storing only the Merkle tree's hash root, we achieve significant reductions in storage space and computational time.

Our design of zero-knowledge proofs incorporates attribute-based commitments. By linking attributes to user addresses, we strengthen the system's defense against man-in-the-middle and replay attacks. Figure 2 illustrates the AH-ZKP structure, encompassing elements like the collision-resistant hash function, pseudo-random sorting function, and the attribute Merkle tree, along with the attribute commitment (COMM) and NP declaration.

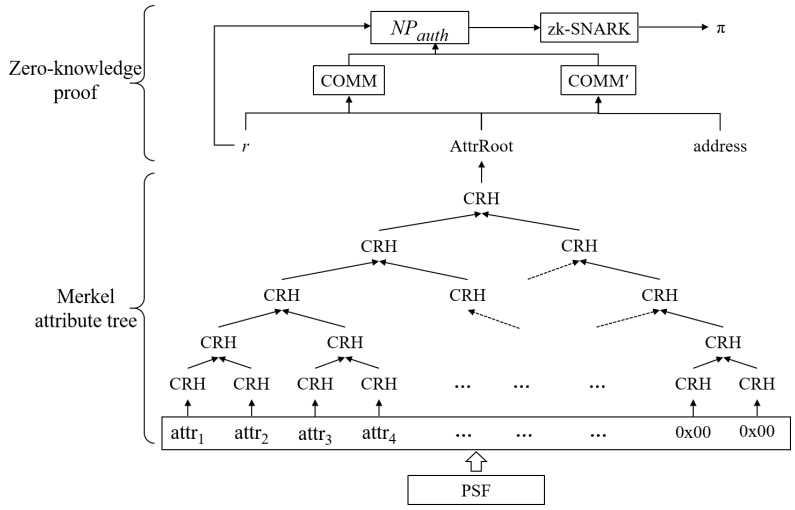

Figure 2. AH-ZKP structure emphasizing attribute hiding.

Key components of the AH-ZKP include the following:

1. Collision-resistant hash function (CRH): We employ the Pedersen hash, which is reliant on the discrete logarithm problem, to ensure anti-collision properties and compatibility with the R within a first-order constraint system (R1CS) required by AH-ZKP.
2. Pseudo-random sort function (PSF): This function is designed for anti-collision pseudo-randomness, ensuring distinct outputs for different inputs while maintaining consistency for identical inputs. The PSF plays a crucial role in obfuscating the attribute list (*AttrList*), including sorting and padding operations.
3. Attribute Merkle tree: Constructed using CRH, this tree provides a quick verification mechanism for attributes within *AttrList*. It stores the hash values of individual attributes in leaf nodes, with missing nodes filled with zero bytes, optimizing storage by retaining only the root hash value (*AttrRoot*).
4. Attribute commitment COMM: The value of $COMM_{attr}$ is computed by COMM based on the attribute root *AttrRoot* and the random number r. *AttrRoot* is the root of the attribute Merkle tree constructed from *AttrList*. In order to check the access rights, the data uploader and the data visitor need to generate the same value of $COMM_{attr}$. The data uploader should secretly send r_{random} to the data visitor in a secure channel. $COMM'_{attr}$ is computed by the COMM based on the attribute root *AttrRoot* and the address $addr_{downloader}$ of the data visitor, thus associating the attribute root *AttrRoot* with the address of the data visitor and preventing an attacker from using $COMM'_{attr}$ to impersonate a legitimate data visitor through a replay attack.
5. NP Statement: The earlier proposed NP_{auth} demonstrates that a data visitor possesses the requisite access rights. This statement includes a private input *AttrList*, public input $addr_{downloader}$, and r_{random}, leading to the generation of $COMM_{attr}$ and $COMM'_{attr}$ through non-interactive commitments.

The conversion of NP_{auth} into a relationship-generating meta R is expressed as

$$R = \left\{ \begin{array}{l} (addr_{downloader}, r_{random}, COMM_{attr}, COMM'_{attr}; AttrList) : \\ COMM_{attr} = COMM(AttrList, r_{random}) \hat{} COMM'_{attr} \\ = COMM(AttrList, addr_{downloader}) \end{array} \right\} \quad (5)$$

We confine R within a first-order constraint system (R1CS), employing it to construct zk-SNARK commitments that underpin the aforementioned functionalities. This ensures that verifiers recognize the data visitor's compliance with access permissions without gaining knowledge of specific attributes possessed.

Our framework for attribute-hiding zero-knowledge proofs (AH-ZKPs) is meticulously architected to enable privacy-preserved authentication within virtual power plant systems. The process commences with an attribute list $AttrList$, strategically obfuscated via a pseudo-random sort function (PSF). We define f as an abbreviation for the collision hash function CRH. Each attribute $attr_i$ undergoes a cryptographic transformation through a collision-resistant hash (CRH) function, yielding hashed counterparts $leaf_i$:

$$CRH(attr_i) = leaf_i, \quad (6)$$

Subsequently, these leaf nodes are systematically aggregated to construct the Merkle tree, employing recursive hash pairings until the Merkle root $AttrRoot$ is derived, encapsulating the entirety of the user's attributes in a single hash value. The integrity of the Merkle tree is paramount, as it serves as the backbone for the zero-knowledge proof by succinctly summarizing the attribute set.

In the ensuing phase, we engender two commitments, $COMM$ and $COMM'$. These are formulated by intricately binding the $AttrRoot$ with a nonce r and the prover's address, expressed as

$$COMM = f(AttrRoot, r), \quad (7)$$

$$COMM' = f(AttrRoot, \text{address}), \quad (8)$$

Integral to the zero-knowledge aspect, these commitments lead to the generation of a non-interactive proof π within the zk-SNARK protocol. This proof robustly asserts the NP statement NP_{auth}, which allows the verifier to ascertain the authenticity of the attribute claims without gaining access to the attributes themselves:

$$NP_{auth} : \{(COMM, COMM'; AttrList) | \exists r, \text{address}\}. \quad (9)$$

A verifier, upon evaluating π, can confirm that the prover has satisfied the prescribed access policy. This process is executed without revealing any specific details of the attributes, thus upholding privacy. Our AH-ZKP framework is a testament to the harmonization of robust security protocols with the imperative of privacy, setting a new benchmark in the field of secure data access for virtual power plant systems.

The following will provide a detailed explanation of the specific steps for using AH-ZKP in conjunction with the plan.

3.5. Specific Steps

Our plan includes a system initialization process and five main steps: upload, verify, download, decdata, and audit. The main process is described as follows.

1. System initialization. We use data attribute-based encryption (DABE) to encrypt the raw data (RawData) and use AH-ZKP to protect attribute privacy. In DABE, the data are divided into multiple parts and are encrypted and stored on different distributed nodes. Each node only stores a portion of the data, and decryption of the data requires certain attribute conditions to be met. This design distributes data storage and management across multiple nodes, improving data security and reliability. The registration and authentication node (RAN) is used to initialize the public security parameters and keys required for DABE and AH-ZKP, while each

attribute management node (AMN) initializes and generates the corresponding public–private key pairs.

2. Upload. The data uploader uploads the file data after encrypting them using the DABE algorithm. During encryption, it is necessary to determine the file's access policy $AttrList$. We use a non-interactive commitment scheme COMM to generate commitments. This means that, given a random number r_{random} and the secret information $AttrList$, a commitment can be calculated. At the same time, others do not know the content of $AttrList$. When given the random number r_{random} and the secret information $AttrList$, anyone can verify and confirm their equality. The data uploader sends the commitments and encrypted data (EncData) to the virtual power plant. The system then sends the data to the data center and obtains a storage address. Subsequently, the storage address and the corresponding attribute commitments are sent to the access control list (ACL) to record the upload behavior. The storage address is then returned to the data uploader.
3. Verify. The data uploader uses AH-ZKP for permission verification. First, they need to generate attribute commitments $COMM_{attr}$ based on their own attributes. Then, the uploader also needs to calculate address commitments based on their own addresses, binding the attributes to the user's address. Finally, the uploader needs to prove the following NP authorization statement NP_{auth} to the terminal using AH-ZKP:
   I have private inputs, an attribute list $AttrList$, $addr_{downloader}$, and a random number r_{random}, and have obtained $COMM_{attr}$ and $COMM'_{attr}$ through non-interactive commitment.
   We transform the above statement NP_{auth} into a relation generating element R and constrain it to R1CS. We use the Groth16 algorithm to generate the AH-ZKP proof of commitment. The data uploader then forwards it to the node to prove their possession of the relevant attributes. The system will verify the correctness and, upon successful verification, query the ACL to obtain the storage address corresponding to $COMM_{attr}$. Subsequently, the node will send the storage address of the encrypted data and the permission credentials for accessing the encrypted data to the data uploader and generate a permission authentication record. The data uploader can then obtain the encrypted data from the data center using the permission credentials.
4. Download. When a data accessor downloads data, they send the storage address, the hash value of their own address, and the verified credentials to the data center. Additionally, they also need to send the usage record encrypted with the data uploader's public key. This record combines the assessor's identity with their address. After the data center verifies the credentials (which prove that the accessor has the necessary permissions to access the address), the encrypted data (EncData) is returned to the data accessor, and a download record is sent.
5. DecData. After receiving the ciphertext, the data accessor sends the uid and attributes to the relevant attribute management node (AMN). Then, the AMN uses the secret key SK to generate the corresponding $K_{i,uid}$ for the attribute i and uid and returns it to the accessor. With sufficient $K_{i,uid}$, the accessor can then decrypt the ciphertext using the DABE algorithm and obtain the original data $RawData$.
6. Audit. When there is an anomaly with the data, the data uploader or relevant nodes can obtain the data download record by reviewing the nodes and decrypt it using the private key SK. Then, they conduct an operational review based on the record to determine the user responsible for the relevant operations.

4. Security Analysis

4.1. Security Model

Our security model for the virtual power plant system is underpinned by several crucial assumptions. Foremost is the trust in the registration and authentication nodes at the system's inception, ensuring the integrity of vital elements like the global parameters and the global identifier uid, which is essential for AH-ZKP and DABE.

The attribute management node (AMN) too plays a pivotal role during system initialization. It is not just about attribute distribution and key generation; the AMN is instrumental in reinforcing DABE's anti-collusion capabilities, preventing collusion attacks by disallowing the combination of attributes from different management sections to satisfy access policies.

We also rely on the robust anti-collision properties of the collision-resistant hash function (CRH) and pseudo-random sort function (PSF), which are the bedrock of our system's security.

However, certain threats remain:

Threat 1: The risk of unauthorized access to encrypted data.

Threat 2: The possibility of attackers decrypting original data from DABE-generated ciphertext.

Threat 3: The danger of attackers inferring private attribute data through analysis of data operation records.

Threat 4: The vulnerability to inference attacks that might expose file access policies.

Considering the security model of our DABE approach, we recognize the registration and authentication nodes, alongside the attribute management node, as trusted entities. Consequently, the DABE's encryption mechanism inherently safeguards against unauthorized data and privacy breaches. Specifically, only individuals possessing relevant attributes, directly countering Threat 2, can decrypt the original data.

In our data sharing and access control framework, user authentication via AH-ZKP is paramount. This process ensures that only verified users can retrieve data storage addresses, effectively addressing Threat 1 by restricting data access to intended recipients only.

Moreover, our system architecture necessitates fulfilling two critical security requirements: attribute indiscernibility and resistance to attribute tampering. These requirements, addressing Threat 3 (attribute privacy) and Threat 4 (access policy privacy), respectively, also substantiate the efficacy of AH-ZKP in our model.

We define our scheme $S = (Upload, Verify, Download, DecData, Audit)$ with the premise that, if it successfully meets both attribute indistinguishability and tamper resistance criteria, it is deemed secure.

We further elaborate on these two pivotal characteristics. They are conceptualized as interactive games between an adversary 'A' and a challenger 'C'. Within each game, the trusted entity operates a system model responding to various requests (upload, verify, download, decdata, and audit) through scheme S. 'A' issues query requests to 'C', who then facilitates these requests via S, ensuring responses are relayed back to 'A'. This process allows 'A' to comprehend query inputs and analyze corresponding responses, which is integral to maintaining the robustness of our security model.

Enhanced discussion on attribute indistinguishability (addressing Threat 3): In our secure system model, the adversary, denoted as A, is limited to accessing only publicly available information. This set includes the general public parameter (GP), the data visitor's address ($\text{addr}_{\text{downloader}}$), the attribute tree root (AttrRoot), non-interactive commitments ($\text{COMM}_{\text{attr}}$ and $\text{COMM}'_{\text{attr}}$), and a random number ($r_{\text{random}}$). The critical aspect of this model is its ability to maintain attribute privacy security. In a theoretical scenario, adversary A's capability to deduce a data visitor's attributes increases post-interaction with our scheme S. Assume A's initial guessing probability is p_1, which, following an adaptive query, improves to p_2. The scheme is deemed attribute indistinguishable when the probability difference, $\Pr[\text{Event}_{\text{Attr}}] = p_2 - p_1$, is negligible.

We define *attribute indistinguishability* via the game *GameI*, which is outlined as follows:

- Challenger C initializes the game by sampling a binary random number $b \in \{0, 1\}$ and setting up two schemes S_0 and S_1, with corresponding system models M_0 and M_1.
- C permits A to query both schemes S_0 and S_1, encompassing operations like upload, verify, download, decdata, and audit. The correspondence between the system models is revealed as $M_{\text{left}} := M_0$ and $M_{\text{right}} := M_1$, albeit in a randomized sequence. A's

objective is to ascertain whether the pairings match: $\left(M_{\text{left}}, M_{\text{right}}\right) = (M_0, M_1)$ for $b = 0$ or $\left(M_{\text{left}}, M_{\text{right}}\right) = (M_1, M_0)$ for $b = 1$.

In essence, our scheme S, encompassing operations (upload, verify, download, decdata, audit), attains attribute indistinguishability in *GameI* for polynomial orders of λ, satisfying the condition

$$\text{Adv}^{\text{I}}_{S,A}(\lambda) < \text{negl}(\lambda) \tag{10}$$

where $\text{Adv}^{\text{I}}_{S,A}(\lambda) = \Pr[\text{GameI}(S, A, \lambda) = 1]$ signifies the statistical advantage of adversary A in *GameI*. Here, $\text{negl}(\lambda)$ represents an inconsequential probability value on a polynomial scale of order λ.

Enhanced discussion on attribute tamper resistance (addressing Threat 4 in AH-ZKP). The concept of attribute tamper resistance is pivotal in ensuring the integrity and security of the AH-ZKP framework, particularly in virtual power plant environments. This resistance is crucial for preventing unauthorized modifications or the exploitation of access policies, thereby safeguarding the privacy of these policies and thwarting attempts by adversaries to misuse privileges. A typical scenario involves preventing an adversary from stealing a private key to decrypt a file without possessing the requisite attributes.

To rigorously define and evaluate this resistance, we introduce an interactive game, *GameII*. In this game, the challenger *C* initializes the AH-ZKP scheme *S* with the necessary parameters and engages with an adversary *A*.

The adversary's objective is to demonstrate the existence of an address addr^*, which, while lacking the requisite attributes for file decryption, can still produce a valid zero-knowledge commitment $\pi_{attr} := \text{AH} - \text{ZKP}(\text{COMM}_{\text{attr}}, \text{COMM}'_{\text{attr}}, r, \text{addr}^*, \text{attr}_i)$, such that $\text{Validate}(\pi_{attr}, K_{veri}) := 1$. The adversary's success in this game would imply a breach in attribute tamper resistance; the scheme's resilience is demonstrated if *A* fails to accomplish this.

Further, the analysis of security and privacy in this scheme is dissected through two distinct proofs, each targeting one core aspect: attribute indistinguishability and attribute tamper resistance. These proofs collectively affirm the robustness of the scheme against both privacy invasion and unauthorized attribute modification, reinforcing the overall security framework of the system in virtual power plant contexts.

4.2. Proof of Attribute Indistinguishability

Theorem 1. *The scheme* $S = (Upload, Verify, Download, DecData, Audit)$ *demonstrates the property of indistinguishability in an adversarial setting, where adversary A is restricted to accessing only public parameters.*

Consider a simulation game G_{sim}, where adversary *A* interacts with challenger *C* in a manner akin to $Game_I$. Unlike $Game_I$, however, each response in G_{sim} to adversary *A* is independent of the binary variable *b*. Consequently, adversary *A* gains no advantage in G_{sim}, as its success probability remains at 0. This setup allows us to establish that any disparity in *A*'s advantage between $Game_I$ and G_{sim} is, in fact, negligible.

Proof of Theorem 1. The simulation game G_{sim} unfolds as per the following protocol:

- Challenger *C* initializes the essential parameters and disseminates the public parameters to adversary *A*, ensuring the values are predetermined and consistent.
- In G_{sim}, the AH-ZKP key is initialized using $sim()$, in contrast to the standard $Set - up()$, underlining a fundamental divergence in the simulation process.
- Challenger *C* establishes instances of the scheme (S_0, S_1) for the subsequent interaction.

□

Subsequently, *C* presents two models, M_{left} and M_{right}, to adversary *A*. Adversary *A* then forwards two query types (Q, Q') to *C*, who responds in a manner contingent upon the query type. The intricate process is detailed below, emphasizing the crucial phases of upload, validation, and so forth.

In the upload phase, we assume Q and Q'.

$$\begin{aligned} Q &= \left(Upload; GP, RawData, PK_{downloader}, PK, addr_{uploader}, attr_i\right) \\ Q' &= \left(Upload; GP, RawData, PK_{downloader}, PK, addr_{uploader}, attr'_i\right) \end{aligned} \tag{11}$$

Challenger C modifies the computation method before returning the results to A. The revised calculation is explicated as follows:

(a) $AttrList = PSF(str_i)$: A random string of appropriate length is used to compute $AttrList$, diverging from the standard computation of $PSF(attr_i)$, where $attr_i$ represents an attribute required by the data visitor.

(b) $AttrRoot = BuildMerkleTree(str)$: Here, a random string str substitutes $AttrList$ in the computation of $AttrRoot$.

(c) $COMM_{attr} = CRH(str, r)$: A similar approach is adopted, where str, instead of $AttrRoot$, is used in the computation of $COMM_{attr}$.

This modified computational approach is also applied to request Q'.

In the validation phase, the response to Q is altered as follows:

(a) $AttrList = PSF(str_i)$: This step mirrors the upload phase, reinforcing the consistency across different phases of the simulation.

(b) $AttrRoot = BuildMerkleTree(str)$: Replicating the approach in the upload phase for computational consistency.

(c) $COMM_{attr} = CRH(str, r)$: Maintaining the established pattern of computation.

(d) $COMM'_{attr} = CRH(str, addr_{downloader})$: A deviation in the computation process by using str instead of $AttrRoot$.

(e) $\pi_{attr} = sim(COMM_{attr}, COMM'_{attr}, r, addr_{downloader}, attr_i)$: The simulation employs analog arithmetic instead of AH-ZKP for proof generation.

(f) $b_2 = sim(\pi_{attr})$: Analogous to the previous step, analog arithmetic substitutes the standard $Validate(\pi_{attr})$ process to generate judgments.

Challenger C applies identical calculations for request Q'.

The download, decryption, and review phases remain consistent with those in $Game_I$, ensuring uniformity across the entire simulation process.

To conclude, G_{sim}, defined above, is juxtaposed with the original *GameI*, termed G_{real}, to demonstrate that the responses in G_{sim} are independent of b, thereby rendering A's advantage in G_{sim} as zero. Proving the negligible difference in A's advantage between G_{sim} and G_{real} effectively substantiates the indistinguishability property of the scheme.

The following discusses the *GameI*-based modification games G_1, G_2, and G_3 played by challenger C and adversary A.

Define Adv_{G_i} as the advantage of A over G_{real} in the game G_i. We use q_{Up} and q_v to denote the number of upload and verification queries issued by opponent A and Num_{attr} as the number of attributes owned by the data visitor. Define Adv_{PFS} as the advantage of A in distinguishing between normal PSF and random input PSF, and Adv_{CRH} as the advantage of A in distinguishing between the normal CRH and the random input CRH.

The difference between G_{sim} and G_{real} is that PSF, AttrRoot, COMM, and AH-ZKP are different. Four progressive games will be used to demonstrate that A has a negligible advantage in discriminating attributes between G_{sim} and G_{real}.

(1) Game G_1. G_1 replaces the AH-ZKP process of G_{real} with a simulation calculation. Specifically, C uses $sim(\varphi)$ instead of $KeyGen(\varphi)$ to generate K_{prov} and K_{veri}. In the verification phase, C computes $\pi_{attr} := sim(info)$ instead of using the correct method. Since there is no regulation and AH-ZKP has perfect zero-knowledge, it is known that $\mathrm{Adv}_{G_1} = 0$.

(2) Game G_2. G_2 changes the parameters used by G_1 in PSF by replacing the $attr_i$ of the input PSF with a random string str_i of appropriate length, i.e., $AttrList := PSF(str_i)$. Each $attr_i$ replacement will generate Adv_{PFS}. When the scope is extended to Num_{attr} attributes, q_{Up} upload queries, and q_V validation queries, the advantage difference between G_2 and G_1 is

$$\left|\mathrm{Adv}_{G_2} - \mathrm{Adv}_{G_1}\right| = (q_{Up} + q_V) * Num_{attr} * \mathrm{Adv}_{\mathrm{PSF}} \tag{12}$$

(3) Game G_3. G_3 changes the parameter used by G_2 in the *BuildMerkleTree* function by replacing the AttrList input to *BuildMerkleTree* with a random string *str* of suitable length, i.e., $AttrRoot := BuildMerkleTree(str)$. Since *BuildMerkleTree* is implemented with a CRH in this paper's scenario, each *AttrList* substitution will produce Adv_{CRH}. In this process, entropy smoothing and collision-resistance properties also play an important role. By replacing *AttrList* with a random string *str*, additional randomness is introduced, increasing the randomness and unpredictability of the output of the cryptographic algorithm, thereby achieving the effect of entropy smoothing. This helps to improve the security of the cryptographic algorithm, preventing the regularity of input data from affecting the randomness of the algorithm's output. At the same time, since a CRH is used as the hash function for building the Merkle tree, replacing *AttrList* each time will result in Adv_{CRH}. This indicates that replacing input parameters affects the output of the CRH, thereby influencing the construction process of the Merkle tree and the final hash value. Maintaining the collision-resistance property of the CRH is crucial to ensure that different input data will not produce the same hash value after hashing, thus ensuring data integrity and security. When the scope is extended to Num_{attr} attributes, q_{Up} upload queries and q_V validation queries, the advantage difference between G_3 and G_2 is

$$\left|\mathrm{Adv}_{G_3} - \mathrm{Adv}_{G_2}\right| = (q_{Up} + q_V) * (2 * Num_{attr} - 1) * \mathrm{Adv}_{\mathrm{CRH}} \tag{13}$$

(4) Game G_{sim} . G_{sim} changes the parameter used by G_3 in COMM by replacing the *AttrRoot* of the input COMM with a random string *str* of suitable length, i.e., $AttrList := COMM(str)$. Since *BuildMerkleTree* is implemented with a CRH in the scheme of this paper, each *AttrRoot* substitution will generate Adv_{CRH}, and the upload step will generate one time and the verification step will generate two times. When the scope is extended to q_{Up} upload queries and q_V validation queries, the advantage difference between G_{sim} and G_3 is as in Formula (14).

$$\left|\mathrm{Adv}_{G_{sim}} - \mathrm{Adv}_{G_3}\right| = (q_{Up} + 2 * q_V) * \mathrm{Adv}_{\mathrm{CRH}} \tag{14}$$

In summary, the advantage of adversary A in discriminating properties between G_{sim} and G_{real} is as in Formula (15).

$$\begin{aligned} \left|\mathrm{Adv}_{G_{real}} - \mathrm{Adv}_{G_{sim}}\right| &\le \mathrm{Adv}_{G_1} + \left|\mathrm{Adv}_{G_2} - \mathrm{Adv}_{G_1}\right| \\ &+ \left|\mathrm{Adv}_{G_3} - \mathrm{Adv}_{G_2}\right| + \left|\mathrm{Adv}_{G_{sim}} - \mathrm{Adv}_{G_3}\right| \\ &= (q_{Up} + q_V) * Num_{attr} * \mathrm{Adv}_{\mathrm{PSF}} \\ &+ \left[(q_{Up} + q_V) * (2 * Num_{attr} - 1) + (q_{Up} + 2 * q_V)\right] * \mathrm{Adv}_{\mathrm{CRH}} \end{aligned} \tag{15}$$

Furthermore, due to the response value's independence from b in G_{sim}, its dominance is zero, which is deduced considering Equation (6).

$$\begin{aligned} \mathrm{Adv}^{\mathrm{I}}_{S,A}(\lambda) &\le (q_{Up} + q_V) * Num_{attr} * \mathrm{Adv}_{\mathrm{PSF}} \\ &+ \left[(q_{Up} + q_V) * (2 * Num_{attr} - 1) + (q_{Up} + 2 * q_V)\right] * \mathrm{Adv}_{\mathrm{CRH}} \end{aligned} \tag{16}$$

Given the unidirectionality of PSF and CRH, both Adv_{PFS} and Adv_{CRH} are deemed negligible on a polynomial scale of order λ . Consequently, $Adv^{I}_{S,A}(\lambda)$ is also negligible at this scale, affirming the indistinguishability property of the scheme:

$$S = (Upload, Verify, Download, DecData, Audit) \tag{17}$$

4.3. Attribute Tamper Resistance Certification

Theorem 2. *Attribute tamper resistance. In the context of our proposed scheme for virtual power plant user data security, attribute tamper resistance is a pivotal property. It implies that an adversary A cannot significantly alter user attributes without detection. Formally, in GameII, a polynomial-*

size game of order λ, our scheme $S = (Upload, Verify, Download, DecData, Audit)$ is deemed tamper-resistant if the following inequality holds true:

$$\mathrm{Adv}_{S,A}^{\mathrm{II}}(\lambda) < \mathrm{negl}(\lambda) \tag{18}$$

Here, $\mathrm{Adv}_{S,A}^{\mathrm{II}}(\lambda) = \Pr[\mathrm{GameII}(S, A, \lambda) = 1]$ *quantifies the probability of A's success in GameII, and* $\mathrm{negl}(\lambda)$ *denotes a negligible function in the polynomial scale of* λ.

Proof of Theorem 2. We categorize the possible successful tampering events by A into disjoint categories to comprehensively argue the tamper resistance of our scheme.

(1) $EVENT_{addr}$: A wins *GameII* with an alternate address $addr'$ equating to the target $addr*$.

(2) $EVENT_{PSF}$: A wins *GameII* where $addr' \neq addr*$. For some attribute $attr_i'$ of $addr'$ and $attr*_i$ of $addr^*$, we have $AttrList' = PSF(attr_i')$ and $AttrList* = PSF(attr*_i)$, yet $AttrRoot' = AttrRoot*$.

(3) $EVENT_{root}$: A wins *GameII* with $addr' = addr*$ and $AttrRoot' \neq AttrRoot*$. Here, $COMM'_{\mathrm{attr}'} = CRH(AttrRoot', addr')$ and $COMM'_{\mathrm{attr}*} = CRH(AttrRoot*, addr*)$ are equivalent.

(4) $EVENT_{COMM}$: A wins *GameII* where $addr' \neq addr*$ and $AttrRoot' \neq AttrRoot*$, yet $COMM'_{\mathrm{attr}'} = COMM'_{\mathrm{attr}*}$ remains true. □

In summary, the adversary's advantage in *GameII* is represented as follows:

$$\begin{aligned}\mathrm{Adv}_{S,A}^{\mathrm{II}}(\lambda) = {} & \Pr[EVENT_{addr}] + \Pr[EVENT_{PSF}] \\ & + \Pr[EVENT_{root}] + \Pr[EVENT_{COMM}]\end{aligned} \tag{19}$$

Thus, demonstrating that each event's probability is negligible on a polynomial scale of λ confirms the negligible nature of $\mathrm{Adv}_{S,A}^{\mathrm{II}}(\lambda)$, validating the tamper resistance of our scheme.

(1) **Probability of** $EVENT_{addr}$: Define $\varepsilon_1 = \Pr[EVENT_{addr}]$. In the context of our DABE scheme, ε_1 represents the probability that A can substitute an alternate address $addr'$ for the intended target $addr*$ undetected. Given the robust encryption and verification mechanisms of DABE, this probability is expected to be negligible.

(2) **Probability of** $EVENT_{PSF}$: Let $\varepsilon_2 = \Pr[EVENT_{PSF}]$. This event signifies A's success in finding an alternate address $addr'$ where the attribute lists generated by the pseudo-set function (PSF) for $addr'$ and $addr*$ collide, i.e., $AttrList' = AttrList*$. The negligible likelihood of such a collision, given the collision resistance of PSF, further asserts our scheme's resilience.

(3) **Probability of** $EVENT_{root}$: For this event, define $\varepsilon_3 = \Pr[EVENT_{root}]$. This occurs when A manages to generate a matching attribute root $AttrRoot\prime$ for a different set of attributes, thus defying the unique nature of Merkle tree constructions used in our DABE approach. The improbability of this event, given the cryptographic strength of Merkle trees and collision-resistant hash (CRH) functions, underscores the robustness of the scheme.

(4) **Probability of** $EVENT_{COMM}$: Finally, $\varepsilon_4 = \Pr[EVENT_{COMM}]$ assesses the chance of A successfully creating a collision in the commitment scheme, despite differing attribute roots and addresses. The commitment scheme's reliance on CRH ensures the minuscule probability of such an occurrence, reinforcing the security of our system.

Overall, these probabilities, being negligible on a polynomial scale of λ, validate the assertion that $\mathrm{Adv}_{S,A}^{\mathrm{II}}(\lambda)$ is negligible. Consequently, our DABE scheme ensures attribute tamper resistance, which is crucial for maintaining the integrity and confidentiality of user data in virtual power plants.

5. Experimental Simulation

To assess the efficacy of our proposed scheme, we established a simulation environment on a Lenovo desktop computer produced in Guangzhou, China, which is equipped with an Intel Core i5-10500T CPU @ 2.30GHz and 8GB RAM, operating under Ubuntu 18.04. We selected Java 1.8.0 for implementing the distributed attribute-based encryption (DABE)

due to its robust security features and wide use in cryptographic applications. Similarly, the zero-knowledge proof component was developed using the Trust 1.38.0 nightly framework, which was chosen for its advanced cryptographic capabilities.

Our simulation infrastructure focuses on integrating DABE with zero-knowledge proofs within the AH-ZKP architecture, employing the bellman library for its comprehensive support of the rank 1 constraint system (R1CS). This choice facilitates the imposition of constraints on inputs, which are essential for converting the NP statement NP_{auth} (referenced in Section 3.4) into corresponding R1CS constraints.

Furthermore, AH-ZKP's attribute commitments $\{\pi_{attr}\}$ are disseminated across nodes via the web3 interface, promoting an efficient distribution mechanism. Following the architecture setup, the DABE algorithm was constructed using JDK to ensure compatibility and performance efficiency. The user interaction interface and the encryption/decryption modules were developed in Python and optimized for ease of use and versatility. The comprehensive structure of this simulation setup, including the integration of various components, is elucidated in Figure 3, providing a visual representation of the system's workflow and interconnections.

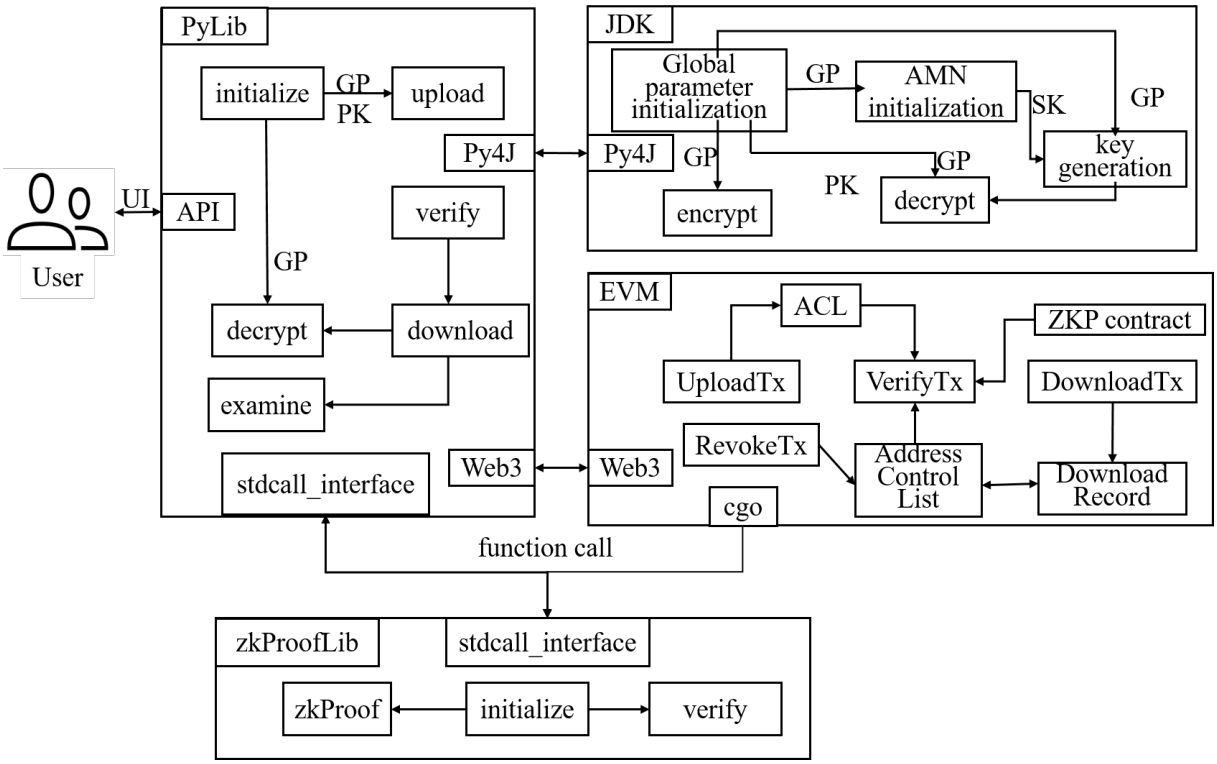

Figure 3. Implementation structure of the plan.

As depicted in Figure 4, an increase in user attributes leads to a corresponding rise in the time cost for both AH-ZKP and traditional DABE methods. However, it is notable that AH-ZKP maintains a significantly lower time cost compared to DABE, indicating enhanced efficiency.

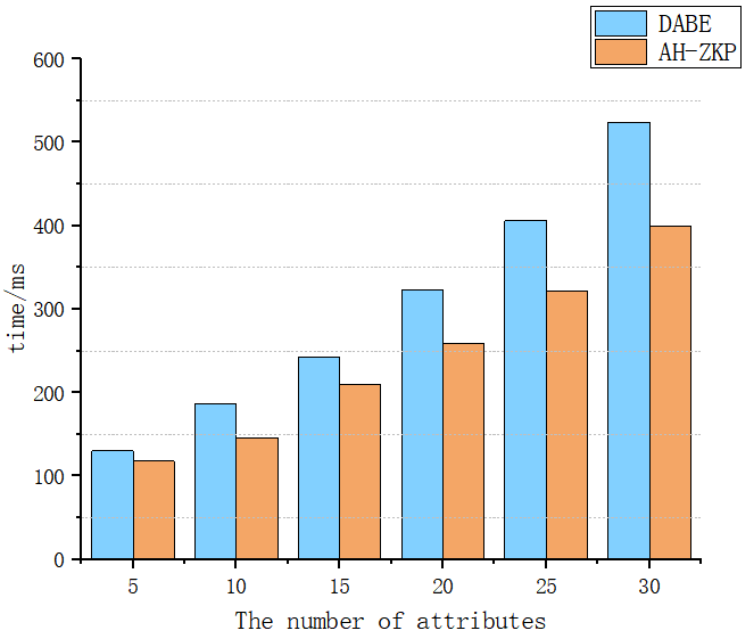

Figure 4. Comparison of time costs between DABE and AH-ZKP algorithms.

Further delving into the performance metrics, Table 3 presents a detailed analysis. Notably, the time efficiency in verifying AH-ZKP is considerably superior to that of any process in DABE, underscoring AH-ZKP's practical applicability in scenarios demanding swift data processing.

The simulation tests utilized two AMNs and five attributes. While the AH-ZKP initialization process incurs a time cost of approximately 6.7 s, it is a one-time process during system initialization. In contrast, DABE requires multiple runs of AMN initialization and key generation for each AMN and specific attributes. Consequently, the cumulative time of AH-ZKP processes, including proof generation and verification, is less than that of the DABE processes, which encompass encryption, key generation, and decryption. Furthermore, the total time cost of DABE escalates with the addition of attributes, thereby making AH-ZKP more time-efficient proportionally. Additionally, as detailed in Table 3, the time cost of other transactions is minimal, further reinforcing the efficiency of AH-ZKP.

Table 3. Time and space costs of each process in the program.

	Procedure	Time Cost	Space Cost
AH-ZKP	Initialization (run only once)	6.7359 s	Attribute parameter 1 KB
	Prove	0.7041 s	Attribute Proof 304 B
	Validate	0.0067 s	/
	Total time	0.7108 s	/
DABE	Global parameter initialization	0.4969 s	/
	AMN initialization (multiple runs required)	0.3518 s	Public key 1 KB Private key 787 B
	Key generation (multiple runs required)	0.0909 s	Attribute key 255 B
	Encryption	0.7602 s	Data size 3 KB
	Decryption	0.1115 s	/
	Total time	1.2351 s	/

In conclusion, the integration of zero-knowledge proofs for privacy protection within the proposed scheme introduces an acceptable overhead, validating its practical feasibility in real-world applications.

6. Conclusions

In this paper, we have developed an innovative privacy protection mechanism for virtual power plants, prioritizing user privacy in varied business contexts. Our approach integrates zero-knowledge proofs with distributed attribute-based encryption (DABE), enhancing both privacy and access control. The DABE framework facilitates file encryption, enabling granular access control while mitigating the risks associated with centralized attribute management through multiple attribute management nodes. This design significantly reduces the likelihood of unauthorized policy inference and ciphertext decryption. Central to our scheme is the attribute hiding zero-knowledge proof (AH-ZKP), which effectively conceals user attributes during verification and authorization. This method ensures the confidentiality of sensitive information, including user attributes and decryption prerequisites, across all transaction stages. We rigorously analyzed the security aspects of our scheme, confirming its robustness in providing attribute indistinguishability and resistance to attribute tampering. Comparative experimental analysis, as shown in our results, demonstrates the scheme's efficiency. Our tests reveal that the attribute-based encryption and zero-knowledge proof steps require approximately 1 s in total computation time, outperforming existing methods in terms of time and space efficiency. The proposed scheme excels in encrypted sharing, multi-party cooperation, collision resistance, verifiability, anonymity, and access policy protection. It proves particularly effective in securing user privacy across various application scenarios within virtual power plants, thereby addressing a critical need in the field.

Author Contributions: Conceptualization, R.Y.; methodology, H.G.; software, F.S.; validation, J.W. and R.Y.; formal analysis, H.G.; investigation, J.W.; resources, F.S.; data curation, H.G.; writing—original draft preparation, R.Y.; writing—review and editing, H.G.; visualization, F.S.; supervision, R.Y.; project administration, F.S.; funding acquisition, R.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been supported by the National Key Research and Development Program of China and Key Technologies for Aggregation and Interactive Control of Scalable and Flexible Virtual Power Plants under grant 2021YFB2401200.

Data Availability Statement: The data presented in this study are available in this article.

Conflicts of Interest: Author R.Y. was employed by the company State Grid Smart Grid Research Institute Co., Ltd. Author J.W. was employed by the company State Grid Shanghai Municipal Electric Power Company. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Hui, H.; Ding, Y.; Shi, Q.; Li, F.; Yan, J. 5G network-based Internet of Things for demand response in smart grid: A survey on application potential. *Appl. Energy* **2020**, *257*, 113–972. [CrossRef]
2. Zou, Y.; Yang, L. Synergetic dispatch models of a wind/PV/hydro virtual power plant based on representative scenario set. *Power Syst. Technol.* **2015**, *39*, 1855–1859. [CrossRef]
3. Lin, Y.; Miao, S.; Yang, W.; Yin, B.; Tu, Q.; Yw, C. Day-ahead optimal scheduling strategy of virtual power plant for environment with multiple uncertainties. *Electr. Power Autom. Equip.* **2021**, *41*, 143–150.
4. Liu, D.; Fan, Q.; You, H.; Dai, X.; Huang, Y.; Shao, Z. Research status and trends of virtual power plants under electrical Internet of Things. *Adv. Eng. Sci.* **2020**, *52*, 10.
5. Zhang, K.; Ding, G.; Wen, M.; Hui, H.; Ding, Y.; Zhu, J.; Xie, K.; Yu, C.; Zhang, L. Review of optimal dispatching technology and market mechanism design for virtual power plants. *Integr. Intell. Energy* **2022**, *44*, 60–72.
6. Fang, L.; Xu, Y.; Yang, X.; Li, L.; Fu, G.; Chai, Z. Multi-time scale coordinated operation strategy of virtual power plant clusters considering power interactive sharing. *Power Syst. Technol.* **2022**, *46*, 642–656.
7. Liu, S.; Ai, Q.; Zheng, J.; Wu, R. Bi-level coordination mechanism and operation strategy of multitime scale multiple virtual power plants. *Proc. CSEE* **2018**, *38*, 753–761.

8. Belguith, S.; Kaaniche, N.; Russello, G. Lightweight Attribute-based Encryption Supporting Access Policy Update for Cloud Assisted IoT. In Proceedings of the 2018 IEEE 11th International Conference on Cloud Computing (CLOUD) San Francisco, CA, USA, 2–7 July 2018, Volume 2, pp. 135–146.
9. Guan, Z.; Yang, W.; Zhu, L.; Longfei, W.U.; Wang, R. Achieving adaptively secure data access control with privacy protection for lightweight IoT devices. *Sci. China Inf. Sci.* **2021**, *64*, 14. [CrossRef]
10. Hur, J.; Noh, K.D. Attribute-Based Access Control with Efficient Revocation in Data Outsourcing Systems. *IEEE Trans. Parallel Distrib. Syst. Publ. IEEE Comput. Soc.* **2011**, *22*, 1214–1221. [CrossRef]
11. Li, M.; Yu, S.; Zheng, Y.; Ren, K.; Lou, W. Scalable and Secure Sharing of Personal Health Records in Cloud Computing Using Attribute-Based Encryption. *IEEE Trans. Parallel Distrib. Syst.* **2013**, *24*, 131–143. [CrossRef]
12. Khalil, A.; Mbarek, N.; Togni, O. IoT-MAAC: Multiple Attribute Access Control for IoT environments. In Proceedings of the 2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC), Vegas, NV, USA, 10–13 January 2020.
13. Hayata, J.; Ishizaka, M.; Sakai, Y.; Hanaoka, G.; Matsuura, K. Generic Construction of Adaptively Secure Anonymous Key-Policy Attribute-Based Encryption from Public-Key Searchable Encryption. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **2020**, *E103.A*, 107–113. [CrossRef]
14. Li, J.; Ai, Q. Operation mode of virtual power plant considering peak regulation auxiliary service. *Electr. Power Autom. Equip.* **2021**, *41*, 7.
15. Zhu, Y.; Yi, Z.; Lu, Q.; Yang, Y.; Li, B.; Xu, Y. Collaborative pricing strategy of virtual power plant and distribution network considering typical scenes. *Electr. Power Constr.* **2019**, *40*, 74–85.
16. Zhong, Y.; Sun, Y.; Xie, D.; Zhai, S. Multi-scenario optimal dispatch of integrated community energy system with power-heating-gas-cooling subsys-tems. *Autom. Electr. Power Syst.* **2019**, *43*, 9.
17. Chen, W.; Sun, R.; Qiu, J.; Chai, Q. Profit allocation and frequency regulation bidding strategy of virtual power plant considering battery cycle life. *J. Glob. Energy Interconnect.* **2020**, *11*, 374–384.
18. Yi, Z.; Xu, Y.; Gu, W.; Wu, W. A Multi-Time-Scale Economic Scheduling Strategy for Virtual Power Plant Based on Deferrable Loads Aggregation and Disaggregation. *IEEE Trans. Sustain. Energy* **2020**, *11*, 1332–1346. [CrossRef]
19. Lv, M.; Lou, S.; Liu, J.; Wu, Y.; Wang, Z. Coordinated optimization of multi—Type reserve in virtual power plant accommodated high shares of wind power. *Proc. CSEE* **2018**, *38*, 9.
20. Qin, Y.; Ge, L.; Bo, W. Swarm intelligence collaborative control and optimiza-tion technology of Energy Internet. *Huadian Technol.* **2021** , *43*, 1–13. [CrossRef]
21. Groth, J. On the Size of Pairing-Based Non-interactive Arguments. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, 8–12 May 2016.