Data Privacy in the Smart Grid: A Decentralized Approach

Angela Upreti
University of Massachusetts
aupreti@cs.umass.edu

Judith B. Cardell
Smith College
jcardell@smith.edu

Dominique Thiebaut
Smith College
dthiebaut@smith.edu

Abstract

Evolution toward the smart grid includes implementation of elements such as smart meters, embedded microprocessors, two-way communication systems from consumers to system operators, and automated demand response as supported through dynamic pricing. Dynamic pricing throughout the smart grid will require frequent transfer of energy consumption data from the customers to the ISOs. Privacy and security issues related to transferring this data are widely studied. However, typical frameworks rely on a trusted third party, such as the ISO or a load aggregator, that would then have access to all of the consumer data.

This paper proposes a Bitcoin-like decentralized model as a solution for secure information transfer within the smart grid, eliminating the presence of a centralized data aggregator or other third party operator. Each smart meter participates as an equal peer in the proposed peer-to-peer network, and elements of authentication, confidentiality and data verification are developed similar to the existing Bitcoin framework. The contribution of this paper is the proposed framework for the smart grid which cryptographically secures the transfer of energy consumption data while ensuring privacy.

1. Introduction

A power system consists of entities that generate, transmit, distribute and consume energy. The consumers in the power system model can be categorized into five categories: residential, commercial, industrial, municipal and agricultural. In the traditional model, consumers and producers of electricity are two distinct components of the power system. More recently, a transactive energy model allows incorporation of entities that both produce and consume electricity. In addition to using energy supplied by the power system, home or business owners can also supply energy to the grid if they are able to produce electricity on site. Increasingly, they are sought after to be active participants in terms of responsive demand that can help in both lowering prices and maintaining reliability.

1.1. The Smart Grid

A power grid is referred to as a *smart grid* when the functionalities of entities of a power grid include embedded microprocessors, two-way communications and improved automation. It is a term that encapsulates the addition of different forms of smart elements to the traditional power grid infrastructure [1]. Remote monitoring of generation, transmission and distribution with communication technologies, negawatt contribution by prosumers to the grid, addition of renewable energies, automated demand response (ADR) all fall under the umbrella the of smart grid. The addition of any element that can support the power system to respond to consumer demand or support the consumer in responding to the conditions on the grid can make the grid "smart."

The traditional power grid does not support two-way communications between the system operators and the consumers. Traditional meters do not record details of hourly or minute-by-minute power consumption with the result that neither the ISO nor consumers have granular (minute-by-minute, hourly) power consumption data, and in addition consumers are unaware of the varying cost of production of electricity.

At present, though the cost of production of electricity increases with increased demand, this change in cost of generation is not reflected in the prices seen by the consumers. Adding demand response to the evolving smart grid can be facilitated by charging the consumers the market price. Consumers will pay a different price per unit of electricity depending on the time of the day and system state. A smart grid and advanced metering infrastructure, AMI, makes such dynamic pricing possible.

A major future goal for the smart grid is to

URI: https://hdl.handle.net/10125/59786
ISBN: 978-0-9981331-2-6

eventually deploy dynamic pricing as an incentive for the consumers to change their energy consumption behavior and so add demand response to the consumer's role in the smart grid [2].

1.2. Privacy and Security

Demand response in the smart grid involves transfer of personal information related to energy consumption, making security and privacy concerns crucial considerations for the smart grid. Privacy and security are two distinct yet interdependent concepts and the details specific to smart grid are discussed next.

Privacy usually means one's ability to make independent decisions about whether or not to disclose information about oneself. Privacy also involves making decisions about who can have access to one's information and who cannot. Often, privacy is achieved through statistical methods of data aggregation. Security on the other hand involves putting physical, technical and network safeguards in place. Securing data means protecting the data from potential attackers and protecting data from being misused. Security is viewed as a technical issue that can be solved using cryptographic protocols.

Security is necessary to achieve privacy. Security is the action, and privacy is the result. In the context of data transmission between smart meters and the utilities, a network attack to the security might compromise privacy. If the cryptographic protocols used in the network are not secure enough, personal data and consequently privacy may be compromised. However, some degree of privacy may be still preserved if strong mathematical tools are used for data aggregation. In a nutshell, strong security can protect privacy and using privacy-enhancing statistical tools can take us a step further in preserving consumer privacy.

This paper proposes a decentralized solution to privacy and security issues in the smart grid [3]. Section 2 highlights the need for granular consumption data along with dynamic pricing. Privacy and security issues are discussed in the context of this granular energy-consumption data.

The following section discusses cryptographic protocols relevant to this paper and the framework for Smart Grid communications proposed here. The Solution section presents a novel decentralized solution for data exchange within the Smart Grid, which borrows from the architecture of the Bitcoin network.

2. Motivation

The smart grid with effective demand response in place can play an important role in decreasing system peaks as well as responding to alert and emergency operating states. The goal of demand response systems is to modify consumption behaviors by using fine-grained pricing information [4].

2.1. Need for Granular Data

To achieve a flatter demand curve for electricity, the ISOs will need to regularly distribute pricing information to the consumers. Pricing information can be communicated to the consumers using smart meters or other forms of gateway devices depending on the architecture chosen for data communication. The consumers can then alter their electricity consumption behaviors based upon the prices they see. Demand response on the consumer's part can be automated or manual. The time interval at which the new price is calculated and sent to the customers can be hourly, every 30 minutes, every fifteen minutes or even minute-by-minute.

We note that fine-grained details about how *each* consumer uses energy is not necessary for the smart grid to operate efficiently. Rather granularity in *aggregate* energy usage per unit time is what is required. We can look at collection of energy consumption data as an action and distribution of pricing information as the goal of the action.

2.2. Privacy and Security Mechanisms

To implement demand response, energy consumption data needs to be transmitted to load aggregators and/or system operators. Securing this data transmission channel for energy consumption data is important for protecting consumer privacy. For example, behavior extraction algorithms can be used by an adversary to obtain information about different appliances used by the consumers [5]. With this information, the adversary can infer whether or not a consumer is home, which room the consumer is occupying and what the consumer is doing while at home. Consequently, in addition to the significant loss of privacy, poorly secured data can can result in physical security risks. Also, if home area network is compromised, an attacker could control electrical appliances within a consumer's home.

This paper specifically addresses securing the communication channel between the gateway devices and the ISOs. Security issues for the gateway devices or the HAN are not considered in the proposed solution.

3. Background

3.1. Security and Privacy Approaches

Several approaches have been suggested to tackle various aspects of privacy preservation in the smart grid. Some widely-discussed solutions to privacy and security issues in the smart grid are listed below[6]. Only some of these tools have been carried over to the decentralized solution presented in this paper.

Escrow Services. These services improve anonymity of Bitcoin transactions. Though identities of transactions are public encryption keys, the movement of these transactions is traceable using data clustering algorithms [7]. Clustering algorithms can also link different public keys related to the same user. Escrow services improve anonymity through shuffling transactions between different users while maintaining the correct Bitcoin amount to and from each user, thus protecting against tracking and clustering. For the framework proposed in this paper, the energy consumption data will include public encryption keys of the user as a part of the header information. Trusted escrow services can randomize these IDs and related data before sending them over to the utilities.

Filtering. Each appliance used in a consumer's home has a certain load signature. If the low frequency components are filtered and blocked, behavior extraction algorithms will not be able to extract useful details about consumer behavior [6]. The framework proposed in this paper assumes that only the energy consumption data broadcast into the network is filtered. The data recorded by the smart meter for billing cannot be filtered however, because consumers need to pay for the total number of kWh they use.

Data Aggregation. Data aggregation us a powerful tool for protecting consumer privacy in the smart grid. Aggregation can happen at different levels: Home Area Network (HAN), Building Area Network (BAN) and Neighborhood Area Network (NAN)[8]. A privacy concern for the smart grid arises when load aggregators have access to identifiable consumer data. The decentralized model proposed in this paper helps to mitigate this risk.

Differential Privacy. Differential privacy is a statistical method of protecting consumers' privacy by making the effects of an individual's data on a distribution indistinguishable. Differential privacy is superior to a simple data perturbation because differential privacy preserves the statistical properties of a distribution and guarantees better privacy-utility trade-off [9]. In the context of a smart grid, energy consumption data can be made differentially private by load aggregators before sending the data to the ISOs.

Cryptographic Protocols. Cryptographic protocols are one of the most important security tools. As mentioned earlier, crytographically securing data transfer is the action and the result is consumer privacy. Several popular cryptographic methods such as *Elliptic Curve Cryptography* [10], Secure Hashing Algorithms (SHA) [10] and homomorphic encryption [11] are all possible solutions for the smart grid. These algorithms can help with confidentiality, integrity, end-point authentication and non-repudiation.

While cryptography, if implemented well, is sufficient to preserve privacy, this protection becomes even stronger when multiple guards are in place. Of all the privacy preserving approaches presented above, this paper focuses on cryptographic solutions, discussed next.

3.2. Cryptographic Protocols

Several cryptographic protocols have been proposed for use with the smart grid, although most are targeted towards a centralized architecture. Many of these protocols are also relevant to a decentralized model, and are discussed below.

Homomorphic Encryption. Homomorphic encryption represents a group of semantically secure encryption functions that allow certain algebraic operations on the plain text to be performed directly on the cipher text [11]. Carrying this idea over to the smart grid, energy consumption data can first be encrypted and then aggregated at each level.

Because homomorphic encryption allows the addition of cipher text without the knowledge of plain text, cipher text representing energy consumption by each appliance can be simply added at the smart meter. This will be the first level of aggregation within a home area network. Cipher texts from different homes can be aggregated at the load aggregator without the knowledge of plain text representing power consumption data for each individual house. Homomorphic encryption is powerful because it can provide security via encryption and can guard privacy by allowing aggregation of cipher text.

Attribute-based Encryption and Access Control. One paper related to a decentralized smart grid architecture describes a decentralized architecture for Key Distribution Centers (KDCs) for the smart grid [8]. Attribute Based Encryption (ABE) offers limited data access to different entities of the smart grid. ABE encryption scheme is based on billinear pairings on elliptic curves and can be very useful in access control.

Open Automated Demand Response (OpenADR

2.0). Open Automated Demand Response (OpenADR) standardizes the message format used for ADR so that dynamic price and reliability signals can be delivered in a uniform and interoperable fashion [12]. How the OpenADR messages are used for demand response is governed by HAN protocols at the consumer's discretion.

OpenADR 2.0 is designed to work over the Internet using simple HTTP messages. For security, openADR uses Transport Layer Security (TLS) on both clients and servers. In addition to the TLS, Open ADR Alliance has established their own certificate authority management system with a third party vendor to ensure system-wide secure communication[12]. OpenADR messages can include digital signatures if non-repudiation is desired.

Smart Energy Profile (SEP 2.0). The SEP 2.0 protocol defines the behavior of smart devices and appliances after the gateway device receives an OpenADR message. SEP 2.0 works within a home area network (HAN) and controls the behavior of smart appliances within a HAN [13]. For authentication, SEP enabled devices use elliptic curve cryptography to dynamically establish security keys between communicating nodes.

Similarly to OpenADR 2.0, in this paper we develop a method for sharing demand response messages bidirectionally between the users and the ISOs. For our framework we use OpenADR rather than SEP. However, while OpenADR has client server architecture, the solution presented in the paper is decentralized, drawing upon peer-to-peer models.

3.3. Decentralization

Pseudonymous networks such as Bitcoin or Tor use mathematical principles and algorithms to govern P2P networks in a decentralized framework, ensuring that no central entity has the capability to access user data. The addition of secure hashing and cryptographic methods increases user trust in the networks. This paper proposes and develops a P2P decentralized architecture for the smart grid. Previous work has promoted data transfer taking place between different levels of smart administration such as load aggregators, substations, and operators, but a P2P solution has not been proposed previously.

3.4. The Bitcoin Network

We borrow from the Bitcoin Network the concepts of public encryption keys, distributed consensus, hashing and digital signatures, and we present each one in detail below.

Public Keys as Account Addresses. Public keys serve as the account address of the users in the Bitcoin network and it is recommended that a new public key be used for each transaction. While public keys provide identities for users in the network, nodes themselves have no identities since each node can have many public keys.

Distributed Consensus on Transactions. A key challenge for digital currencies is achieving *distributed consensus* through which honest nodes (non-malicious or faulty) in the network agree upon and validate each transaction; there must be agreement on who owns which Bitcoin and which coins have been spent. For example, if user Alice wants to pay Bob in Bitcoin, she will broadcast a transaction message to all other nodes in the Bitcoin network containing Alice's digital signature as a proof that she herself broadcast the transaction, Bob's public key as the recipient of the Bitcoin, and a hash pointer that points to the previous transactions that show Alice's receipt of the coin that she is paying to Bob.

Many transactions such as the one described above will be broadcast into the network during a period of time. Distributed consensus is achieved when all the honest nodes in the network agree on the transactions that occurred and the order in which they occurred. At any given point in time, a full node has a record (a Bitcoin ledger) of a chain of blocks of transactions on which it has already achieved consensus and a list of outstanding transactions that it has heard but about which it has not yet achieved consensus. Here, a *block* is a record of a series of transactions, and a Bitcoin ledger contains a series of blocks each of which contains a series of transactions, that together form a *block chain*. The block chain is a data structure in which each block contains a hash pointer to the previous block that it extends.

To achieve consensus among the nodes on the record of the transactions, each full node in the Bitcoin network has a record of all the transactions that have ever taken place since the genesis block. In each round, a node is selected from the network to propose the next block in the chain. If the randomly picked node is honest, it will have already checked the validity of the transactions in the block it is proposing. Transactions are validated by checking if the digital signatures are valid and if the transactions being referenced have not been spent already. In case the node that is randomly picked is malicious and not honest, the other nodes will implicitly disapprove the proposed block chain by not extending their existing chain with the newly proposed block. This is possible because all the other nodes that did not get selected have their own list of transactions that they have been recording and validating. If the transactions in

the proposed block cannot be validated by these other nodes in the network, the other nodes will not extend their block chain. This idea of distributed consensus is adapted for the proposed smart grid privacy framework.

Hashing Algorithms and Proof of Work. Hashing algorithms are frequently used to checked message integrity, and determine which node gets to propose the next block in the blockchain ledger. The Secure Hashing Algorithm SHA-256 is used by the Bitcoin network and is also adopted in the decentralized smart grid model proposed in this paper.

The random selection of a node to propose the next block in the chain is done using a SHA-256 hash puzzle. The node that solves this SHA-256 puzzle the fastest broadcasts its solution to all other nodes as a "proof of its work." Other nodes in the network verify the solution to the hash puzzle before accepting the transaction broadcast by the selected node. These nodes show their approval by extending the block chain they have in their own record. As long as more than 51 percent of the total computing power in the network is honest, a malicious node will not be able to affect the integrity of the network, since it is unlikely that any single malicious peer will have enough computational power to beat all of the honest nodes.

Pesudo-random selection with the SHA-256 puzzle is another aspect of the Bitcoin network that used in the proposed model of the decentralized smart grid. Hardware options are presented in [14] and [15] helpful.

Elliptic Curve Digital Signature Algorithm (ECDSA). Digital signatures are used to validate transactions that are broadcast into the Bitcoin network. The digital signature scheme used in the Bitcoin network is based on *elliptic curve cryptography*. The smart grid model we propose in this paper also uses ECDSA to validate energy consumption data [16].

Communication standards such as OpenADR grant third parties (ISOs and load aggregators) access to consumer's energy consumption data. Even though this data may be aggregated, personally identifiable information may still be vulnerable. Individuals running the neighbor aggregators may be able to identify nodes in the same neighborhood based on the load signatures and other heuristics. Attribute-based encryption (ABE) can be useful in this situation for access control. However, entities with correct access policies will still be able to identify the consumers easily in a centralized architecture. Therefore, a more robust way of anonymizing consumer data is necessary because existing communication protocols like OpenADR and OSGP do not guarantee anonymity from aggregators and ISOs.

We propose a framework to ensure privacy for consumer data in a smart grid. A significant obstacle for anonymizing data, however, is that "identitiless" consumers do need to be billed for their *actual* kWh consumption. Data anonymization protocols for the smart grid need to both consider mechanisms to validate that consumers have paid their bills, and also protect consumer identities. In the next section, we present a mathematical protocol that can be used to bill identitiless consumers.

3.5. Billing with Zero-knowledge proofs (ZKP)

A solution to the problem of billing identitiless consumers can be drawn from an alternate digital currency called the Zerocoin, that incorporates escrow services within the coin protocol itself through use of a *zero-knowledge probabilistic proof* to allow mixing of coins to strengthen anonymity [17].

Working Principle of Zero-Knowledge Proof (ZKP). Interactive zero-knowledge proofs can be used to verify whether a consumer has paid a bill without revealing the consumer's identity. Such proofs are "interactive" because the verification of whether the bill has been paid by a particular consumer depends on an interaction between the two parties, the prover and the verifier.

Zero-knowledge proofs enable an entity to prove that it has some knowledge without revealing the knowledge itself [18]. For billing in the smart grid, a zero-knowledge proof with the elliptic curve digital signature algorithm (ECDSA), is simple as well as efficient [19].

The next section presents our model for the smart grid. This model draws from different ideas discussed thus far. The model presented under *Solution* is the contribution of the paper in the field of smart grid.

4. A Decentralized Solution for Security and Privacy in the Smart Grid

The objective of this paper is to develop a decentralized and anonymity-preserving framework for the exchange of energy consumption data in the smart grid. The contribution of this paper is the development of a P2P and Bitcoin-like architecture for the smart grid. A challenge for smart grid evolution is the lack of privacy protections for consumer energy consumption data.

In this section, a privacy-preserving, decentralized model for smart grid data collection is presented. The major elements of the proposed model are presented in Figure 1, that capture's the solution in its entirety. A method to decentralize the smart grid is discussed

below, followed by a discussion of the form of messages that will be sent to the peers within the proposed decentralized architecture. The role of certification authorities is then highlighted, followed by a section discussing how a public ledger of energy usage messages, EUM, will be kept. Next, ideas borrowed from Bitcoin for random node selection and associated incentives are presented, in the context of the smart grid. Zero-knowledge probabilistic proofs for billing anonymous consumers is presented at the end of this section.

4.1. Decentralization

For the purpose of this discussion, power system infrastructure can be thought of as a graph. Nodes (resident or commercial entities with smart meters) in the system are modeled in a way similar to the nodes of a computer network, such as the internet. Decentralization is introduced to the power system by modeling the power system as an overlay peer-to-peer, P2P, network where each commercial or residential entity is a node in a large graph. Instead of the widely proposed client-server communication paradigm between the utilities/aggregators and the commercial or residential consumers, all of the entities of the power grid are seen as peers in the network in our model.

In this decentralized paradigm, no central authority higher up in the supply chain has control over all the customer data in a manner that would allow linking consumers to their energy consumption data. Energy consumption data is kept as a public ledger, as with blockchain, that any user in the network could access. Although accessible by all users, this public ledger can build on experiences with Bitcoin to prevent any user from making associations between specific users and their energy consumption data. Thus, this design will be analogous to a blockchain in the Bitcoin network where the blockchain is a public ledger of anonymous transactions, further discussed below.

4.2. Public Key as Node Addresses

Public accessibility of energy consumption data warrants maximum care be taken to protect consumer anonymity. Similar to the Bitcoin network, public keys are used as node addresses. This allows origin verification for energy consumption data restraining it to approved smart meters within the network.

Nodes are encouraged to change public keys frequently to prevent adversaries from clustering energy data of a single public key.

4.3. Energy Usage Message (EUM) and EUM Validation

We propose that the record of energy consumption by different nodes be maintained in a manner similar to the Bitcoin network. Nodes are configured so that they broadcast data about their energy consumption to all the other peers as frequently as needed. To automate the EUM broadcasting process, gateway devices such as smart meters can be programmed to send out *energy usage messages* (EUMs) regularly at a consistent time intervals. The broadcast EUM contains:

- A time-stamp recording the time when the EUM was created and first broadcast,
- The total energy consumption data for the node in that time-step, and
- A digital signature of the source smart meter. To allow for smaller key size, ECDSA is proposed for the digital signature scheme.

Smart meters broadcasting EUMs need to frequently change their public key that is recorded in the EUMs that they broadcast. Each smart meter is allowed to create as many key pairs as desired by the consumer. These keys need to be registered with a trusted certification authority to ensure only the registered smart meters can successfully broadcast messages into the smart grid network. After receiving an EUM broadcast by a peer, each node validates the received EUM by verifying the digital signature in the EUM as signature of one of the approved smart meters in the network. The digital signature is verified with the help of a certification authority.

4.4. Certification Authority (CA)

In our decentralized model for the smart grid, validity of an EUM and consequently the authenticity of the smart meter broadcasting the EUM, is checked via public keys. Dedicated smart grid certification authority can be put in place to authenticate these public keys so that unregistered meters or adversaries cannot broadcast fake EUMs into the network. All approved smart meters have pre-installed certificates from the power grid certificate authority. Only the public keys belonging to approved smart meters are accepted as a valid EUM by the peer nodes. If an EUM comes from a node without a valid certificate, the peers will not record the message neither will they pass the EUM along to other peer.

In the context of our model, each registered smart meter receives identical, non-transferable

and non-reproducible token signed by the trusted certification authority. This token needs to be presented to the certification authority by the smart meter to receive a certificate for each new key pair that the smart meter generates. The initial tokens to be used for key generation are identical for all smart meters meaning that the certification authorities themselves cannot distinguish between different smart meters. CAs will only be able to verify whether a given public private key pair is valid or not. Therefore, consumer identity is not compromised in this process. A secure connection (SSL or TLS) should be used for all communications between the smart meters and the CAs to protect against man-in-the-middle attacks. For billing, a smart meter records energy consumption data before broadcasting the EUM.

One natural question to ask at this point is what if the consumers with registered smart meters themselves are malicious? One safeguard for this case is to make the smart meter non-tamperable and non-programmable. Any attempts to tamper with the meter should disconnect the meter from the network.

4.5. Public ledger of EUM

Let us assume that our smart grid model sends energy usage data to the utilities every m minutes. All participating peers in the network will keep a record of all the EUMs they have received during an m-minute interval. We call this record of a series of EUMs collected every m minutes an *e-block*, or energy-block. Every time a node receives a valid EUM from its peer, it adds that EUM to the current e-block; an EUM is added only once to avoid duplication. To transfer the e-blocks to the neighbor aggregators, a node from a neighborhood is polled every m minutes with a SHA-256 puzzle. After being polled, the selected node sends an e-block to the neighborhood aggregator. Neighborhood aggregator then forwards the data in the e-block to the utilities. They data in the e-block can be statistically aggregated before being sent to the utilities.

Because we have a P2P paradigm, neighborhood aggregators themselves are a peer in the network and will have their own record of EUMs, and this provides several advantages for our model. Neighbor aggregators nodes can be maintained by the utilities or by other trusted third parties. When an aggregator receives an e-block from the polled node, it can compare the EUMs in the e-block it received against it own record of EUMs. After the neighborhood aggregator forwards an e-block to the utility, it lets the peers in the P2P network know which e-block was forwarded to the utilities by broadcasting the e-block into the network with neigborhood aggregator-marked digital signature. This way, nodes can check an keep of list of EUMs that they had in record that did not make the e-block sent to the utility.

In case an EUM brodcasted during a particular$m$-minute interval does not make the e-block sent to the utility, a few more attempts to broadcast can be made. This will mean that peers will add this EUM to the next e-block. If an EUM does not make any e-block sent to the utilities even after trying for a few more m minute intervals, the message can be discarded because it will no longer be useful for calculating fluctuations in demand every couple of minutes. Time stamp on every message helps decide when to discard a message.

Nodes can discard any e-blocks which they have had in record for more than an hour or so. After an hour of its creation, an e-block's data has most likely reached the utilities already and if not, the data contained in the e-block is no longer useful in calculation of hourly electricity prices. Hence, deleting an e-block an hour after its creation saves the local nodes from having to maintain a large ledger of data.

4.6. Random Node Selection and Verification

Random node selection process in our model is similar to the random node selection process in the Bitcoin network. Every m minutes, a SHA-256, similar to the one in the Bitcoin network, is presented to the nodes in a neighborhood. The puzzle presented in the smart-grid network is also a search puzzle. There is no other way to solve the puzzle except searching the answer space. This "randomizes" the process of node selection. Like the Bitcoin uses a nonce and a service string related to parameters such as the hash of the previous block in its SHA-256 puzzle, smart grid network can use the header information related to the previous e-block that was sent to the utility and a nonce to build its SHA-256 puzzle. This way a consistent SHA-puzzle can be built for all nodes in the network.

The fastest node to solve the SHA-256 puzzle sends the e-block it has in record to the neighborhood aggregator. As illustrated in Figure 1, this message contains the solution to the hash puzzle in addition to an e-block. The neighborhood aggregator first verifies the solution to the has puzzle in the message. If the solution is correct, the aggregator forwards the data in the e-block to the utilities, else another node needs to be polled from the network.

A neighborhood aggregator should keep accepting e-blocks from the nodes until it finds an e-block with the correct hash solution. This way, in case the hash solution of the first node it hears from does not match,

neighborhood aggregators can take the e-block from the next node that solved the puzzle correctly.

The decentralized smart grid protocol in its entirety has been captured by Figure 1.

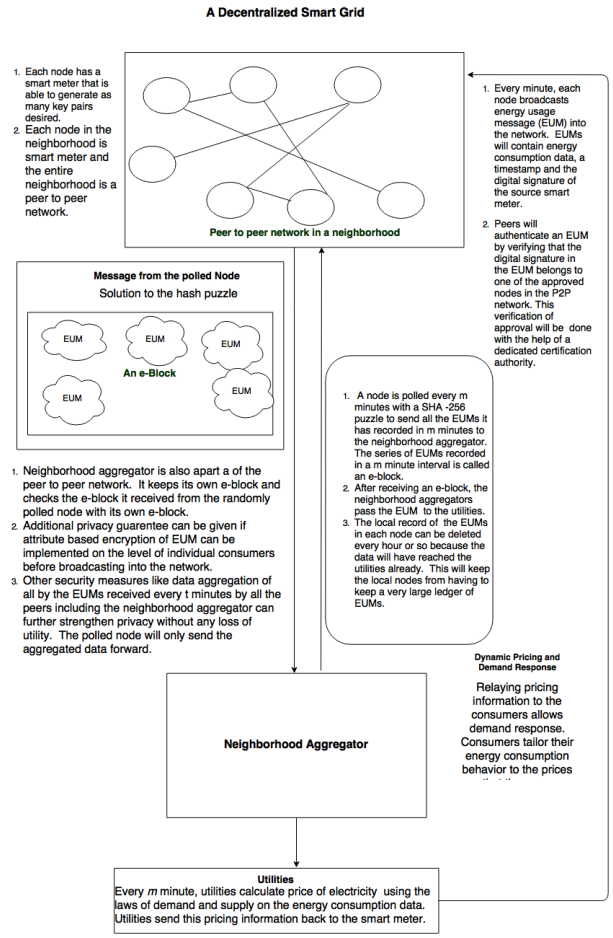

Figure 1. A Decentralized Model for the Smart Grid.

4.7. Incentive for Smart Grid Participation

Some form of incentive is necessary to get the nodes to participate in the hash-puzzle race. Incentives are specially necessary because the nodes will be spending some computational power in trying to solve the puzzle. In our model, smart meters themselves can be the computers that attempt to solve the puzzle. Application specific integrated circuits (ASIC) designed to solve SHA-256 can be incorporated into the smart meter to give them computing power to solve SHA-puzzles.

Like the Bitcoin network rewards 25 BTCs to the first node to solve SHA-256 puzzle, a significant reduction in electricity price can be offered to the first node to solve smart grid hash puzzle. The cost of this reward to the electrical authorities will be offset by the reduction in cost of production brought about by behavioral change due to demand response and dynamic pricing. The expected probabilistic reward needs to be larger in value than the cost of maintaining a Bitcoin node.

4.8. Channel

Similar to Bitcoin, communication between the smart meters can take place over the Internet. Using already existing internet infrastructure for sending energy consumption messages means that the initial start-up cost can be minimal. However, using a dedicated channel could reduce network congestion and latency.

4.9. Connection and Re-connection

The initial connection of any smart meter to the smart grid network can follow the the process of of *Satoshi Client Discovery*. Bitcoin nodes can use one of many potential methods to discover seed peers to join the network [20]. Furthermore, all approved smart meters can come with seed addresses of some peers maintained by the utilities which will help the new smart meters connect to the smart grid network. This way, even if a single node or a portion of the network get disconnected for a while, they can reuse the seed address to get back into the network. Again, our model erases the single point of failure problem.

4.10. Additional Privacy and Security Tools

In addition to the elements already described, more tools can be incorporated into the proposed model to strengthen security and privacy of the consumers. Tools like data aggregation, differential privacy and access control will be discussed in context of our model.

Aggregation. Privacy can be strengthened by making all the nodes participate in data aggregation. As the peers keep receiving EUMs, they can add up the energy consumption data of each EUM they receive. Peers will record this sum of EUMs in their e-block. This way, an e-block does not contain a record of power consumption data per user per minute but will have the total energy consumption data for the entire neighborhood for m minutes. This provides the granularity in time necessary for demand response while improving privacy.

Differential Privacy. Differential privacy can be implemented at the node level in place of data aggregation. Peers will make energy consumption data they have received in m-minute interval differentially private before forwarding the data to the neighborhood

aggregators. To achieve this, synthetic e-block with statistical properties similar to the record of EUMs in the original e-block are generated by each peer node. Each e-block stills contain energy usage per user per minute but contribution of individual nodes remains indistinguishable. This achieves better privacy and more granularity in terms of time as well as the number of users.

Access Control. There exist additional entities in the power grid besides utilities, operators, consumers and neighborhood aggregators. Several different entities might be involved in the process of generation, transmission and the distribution of electrical energy. Attribute-based encryption can be incorporated into our decentralized model for access control among these entities. Before broadcasting an EUM, a node gives attributes to its energy consumption data and encrypt it. For example, data can be given attributes such that it can be decrypted only by the utilities or by operators or the maintenance. This way, not every entity of the grid gains access to all of the data.

4.11. Billing and Zero-Knowledge Proof

For our decentralized model, zero-knowledge probabilistic proofs can be used for billing. In order to use zero-knowledge proof to verify if a consumer has paid their electricity bill, solution x to a discrete log problem $y = g^x \ mod \ p$ can be distributed to those consumers who have paid their electricity bill. Each node will have to acquire a new x every billing cycle and a knowledge of x will prove that the consumer has paid her bill. The verification of payment that is the verification of knowledge of x will be done by the peers.

For verification to be possible, all nodes in a neighborhood will have to have knowledge of the public parameters of some DLP. Neighborhood aggregators can broadcast these public parameters into the smart grid P2P. Same DLP can be used inside a neighborhood but different DLPs should be used across different neighborhoods.

Nodes in a neighborhood can query the peers they are connected to every billing cycle asking for a proof of knowledge of x. Inability of a peer to give a proof of knowledge of x will mean that the peer has not paid her bill. To prevent a node from generating a new address(key pairs) every billing cycle to join the network without paying the bill, certification authorities are set up to check for the solution for the DLP of the last billing before issuing a certificate. In case a consumer has not paid her bill, a message can be sent by peers to the gateway device such as smart meter so as to temporarily disconnect the node from grid. "Zero-knowledge" proof is necessary in this model because disclosing the value of x will mean even the nodes that have not paid their bills will know x.

Care has to be taken because ZKP discussed here is a simplification. Engineering ZKP-based billing system is complex. The implementation of some form of protection against node-collusion is also needed.

5. Conclusion

The objective of this paper is to introduce security and privacy issues related to demand response and smart metering in the smart grid, and propose a decentralized architecture to address these privacy and security issues.

This paper presents the basic elements of a decentralized framework for energy consumption data exchanges in a smart grid. Some potential areas for further research include:

- Congestion control for EUMs during peak times
- Hardware required to solve the SHA-256 puzzles
- The frequency of price calculations and responses of peers in the proposed P2P framework
- Ideal neighborhood size for data aggregation
- EUM message size and required storage capacity

The cost of electricity production varies diurnally and seasonally. Economically efficient consumption of electricity requires relaying prices that reflect the changing cost of production back to the consumer. Use of smart appliances that respond to changing price signals requires availability of a consumer's energy consumption data. This data about consumer's electricity consumption behavior has physical as well as network security risks associated with it. Therefore, anonymizing consumers' energy consumption data can add tremendous advantages to the smart grid. This work set out to solve this problem of consumer data anonymization using the Bitcoin network as a model.

Inspired by anonymous digital currencies such as Bitcoin and Zerocoin, this work established ECC, SHA-256 random selection, sufficient incentives and zero-knowledge proof as some of the important features needed in a decentralized smart grid architecture. The overall architecture of a decentralized smart grid proposed here maps the features in our model to their parallels in the Bitcoin and Zerocoin networks. The proposed framework protects consumer privacy, and promotes security by employing features well tested by the crypto-community. In addition, this model can work over the Internet with means no significant set up cost for addition infrastructure will be required.

6. Acknowledgement

This material is partially based upon work supported by the Department of Energy under Award Number DE-OE0000843.

References

[1] R. H. C. Hertzog, *Data Privacy for the Smart Grid*. Auerbach Publications, Jan. 2015.

[2] C. Brooks, "Smarter metering," *Renewable Energy Focus*, vol. 15, pp. 16–19, Sept. 2014.

[3] A. Upreti, "Security and privacy issues in smart grid: a decentralized approach," May 2016. https://scholarworks.smith.edu/theses/1673/.

[4] S. Wicker and R. Thomas, "A Privacy-Aware Architecture for Demand Response Systems," in *2011 44th Hawaii International Conference on System Sciences (HICSS)*, pp. 1–9, Jan. 2011.

[5] R. O'Donnell, "Prolog to "Privacy-Aware Design Principles for Information Networks"," *Proceedings of the IEEE*, vol. 99, pp. 328–329, Feb. 2011.

[6] S. R. Rajagopalan, L. Sankar, H. V. Poor, and S. Mohajer, "Smart Meter Privacy: A Utility-Privacy Framework," Aug. 2011.

[7] E. Androulaki, G. O. Karame, M. Roeschlin, T. Scherer, and S. Capkun, "Evaluating User Privacy in Bitcoin," in *Financial Cryptography and Data Security* (A.-R. Sadeghi, ed.), no. 7859 in Lecture Notes in Computer Science, pp. 34–51, Springer Berlin, Apr. 2013.

[8] S. Ruj and A. Nayak, "A Decentralized Security Framework for Data Aggregation and Access Control in Smart Grids," *IEEE Transactions on Smart Grid*, vol. 4, pp. 196–205, Mar. 2013.

[9] C. Dwork, "Differential Privacy," in *Automata, Languages and Programming* (M. Bugliesi and et. al., eds.), no. 4052 in Lecture Notes in Computer Science, pp. 1–12, Springer Berlin, July 2006.

[10] S. Nakamoto, *Bitcoin: A peer-to-peer electronic cash system*. https://bitcoin.org/bitcoin.pdf.

[11] F. Li, B. Luo, and P. Liu, "Secure Information Aggregation for Smart Grids Using Homomorphic Encryption," in *2010 First IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pp. 327–332, Oct. 2010.

[12] O. Alliance", "OpenADR: In a Nutshell," Feb. 2016. http://www.openadr.org/assets/docs/DTECH2015/whatisopenadr.pdf.

[13] M. Saxena, "Smart Energy Profile (SEP) 2.0 Uncovered EE Times," *EETimes*, 2016.

[14] "ASIC," May 2015. https://en.bitcoin.it/wiki/ASIC.

[15] "Mining hardware comparison," Sept. 2015. https://en.bitcoin.it/wiki/Mining_hardware_comparison.

[16] D. Johnson, A. Menezes, and S. Vanstone, "The Elliptic Curve Digital Signature Algorithm (ECDSA)," *International Journal of Information Security*, vol. 1, pp. 36–63, Jan. 2014.

[17] I. Miers, C. Garman, M. Green, and A. D. Rubin, "Zerocoin: Anonymous Distributed E-Cash from Bitcoin," pp. 397–411, May 2013.

[18] S. Goldwasser, S. Micali, and C. Rackoff, "The Knowledge Complexity of interactive Proof Systems*," *SIAM Journal of Computation*, vol. 18, pp. 186–208, Feb. 1989.

[19] D. Chaum, J.-H. Evertse, and J. v. d. Graaf, "An Improved Protocol for Demonstrating Possession of Discrete Logarithms and Some Generalizations," in *Advances in Cryptology EUROCRYPT 87* (D. Chaum and W. L. Price, eds.), no. 304 in Lecture Notes in Computer Science, pp. 127–141, Springer Berlin, Apr. 1987.

[20] "Satoshi Client Node Discovery," Mar. 2014. https://en.bitcoin.it/wiki/Satoshi_Client_Node_Discovery.