NILM-Net: Anonymous Coordination of Real-Time Power Monitors for Smart Grid Applications

Aaron W. Langham
Massachusetts Institute of Technology
Cambridge, MA
alangham@mit.edu

John Donnal
United States Naval Academy
Annapolis, MD
donnal@usna.edu

Steven Leeb
Massachusetts Institute of Technology
Cambridge, MA
sbleeb@mit.edu

***Abstract*—Coordination of decentralized energy resources, condition based maintenance, and energy conservation all benefit from visibility of itemized energy demand throughout the power distribution network. Nonintrusive load monitoring (NILM) can provide itemized power demand for an installation but sharing this information without sacrificing privacy is a challenging and unsolved problem. This work presents NILM-Net, a gossip protocol for aggregating itemized load events into an anonymous central database. Unlike traditional gossip protocols, NILM-Net provides monetary incentive for participation and malicious nodes can be efficiently detected.**

***Index Terms*—Smart grid, nonintrusive load monitoring, privacy, decentralized networks, Internet of Things**

I. Introduction

High-quality power monitoring can improve the efficiency, health, and resiliency of the smart grid. By understanding how energy is consumed, producers can schedule generation more efficiently, users can track the maintenance condition of their appliances, and spikes in demand can be predicted and mitigated to avoid reduced power quality or outages.

Distributed energy resources (DERs) have disrupted the traditionally unidirectional power grid [1], [2]. With DERs, microgrids can supplement or replace the centralized utility for a local set of users [3], [4]. These participants in the microgrid's energy market need to accurately forecast demand to maximize revenue. However, this information is typically held by the utility, which may be unwilling or unable to make data available. Even if it were available, the time series of aggregate power consumption is of little use to local microgrid communities. Instead, an energy database can be queried in near real-time to provide historical demand information.

Condition-based maintenance (CBM) promises reduced cost of ownership and decreases the likelihood of unexpected component failure. However, CBM requires real-time knowledge of equipment condition through sensing. Current research focuses on the development of digital twins [5], [6]. Unfortunately, this requires significant modeling and simulation that are not feasible for the majority of residential and commercial appliances. Instead, cohort analysis provides benchmarks on healthy operation at a much lower cost. However, there must be an established method to query load operation in the cohort, which requires some type of centralized energy database.

Finally, itemized power monitoring can improve the resiliency of the grid through energy reduction and more efficient consumption profiles. Comparison metrics between consumers have been shown to be effective in motivating energy reduction [7]. Directly sharing this data between consumers, or more typically with a commercial third party, exposes sensitive information unrelated to the objective of comparative consumption such as patterns of life and even indications of occupant behavior [8], [9]. A database that only holds anonymized load events provides the same value without the corresponding privacy concerns of handling the raw data.

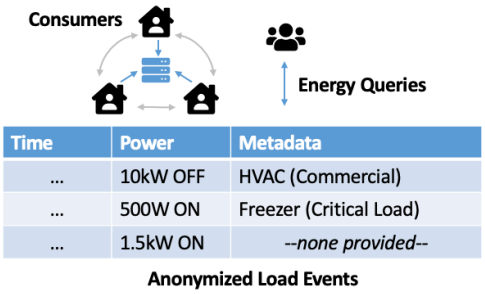

Fig. 1. NILM-Net allows grid participants to dynamically query electricity usage patterns enabling a variety of smart grid applications. Privacy protections as well as monetary incentives encourage data sharing by energy consumers.

In all of these situations a shared database of itemized load operation provides value. Nonintrusive load monitoring (NILM) can generate this data at the site level, but integrating NILM systems at scale to provide grid-wide visibility of demand while maintaining users' privacy has remained an open problem. To address this gap, we present NILM-Net, a protocol to anonymize and consolidate itemized power consumption data into a centralized system that can be queried on demand by grid participants (Figure 1). A monetary compensation scheme encourages data sharing, and a robust fraud detection mechanism identifies invalid or modified data and the malicious node(s) that produced it. The result is an efficient and scalable system that can deliver on the promises of the smart grid to the energy producer and consumer.

II. Background

The current state of the art in commercially deployed power monitoring is Advanced Metering Infrastructure (AMI) [10]. However, AMI produces too much data with too little informa-

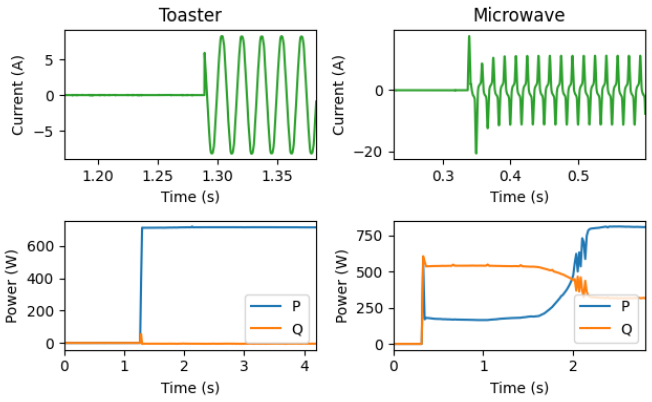

Fig. 2. Active and reactive power turn-on transients for a toaster (left) and a microwave (right). The two loads clearly present different signatures.

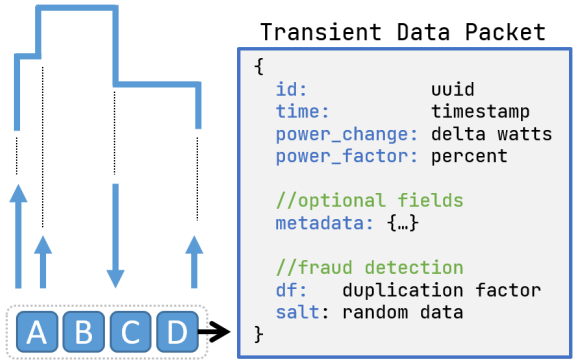

Fig. 3. The NILM-Net packet structure. Aggregate power consumption (top left) is divided into discrete transients corresponding to the operation of individual loads (middle left). Data associated with these transients are composed into packets (bottom left) containing the fields shown on the right.

tion. Sampling intervals on the order of minutes are sufficient to track real-time demand, but is of limited use because the source of the demand is unexplained. For example, cycling systems (e.g. HVAC), appliances with flexible scheduling (e.g. electric vehicle chargers), and appliances with time-sensitive demand (e.g. kitchen appliances) imply very different power requirements but with such a low sampling rate, they are indistinguishable from one another except by their average wattage which is often quite similar. Unlike AMI, NILM can itemize power consumption by appliance. This opens up an entire new field of power system optimization.

NILM exploits Kirchoff's current law by taking aggregate current measurements at the utility service point and then uses disaggregation techniques to itemize power consumption by load [11]. Sufficiently high sampling rates enable recognizable "fingerprint" events in the aggregate power stream. Figure 2 shows an example of such fingerprints for a toaster and a microwave. The toaster, a resistive load, consumes only real power, whereas the microwave consumes both real and reactive power and contains harmonic distortions in the current signal. A NILM system can measure these differences and determine which loads are operational in a particular building or residence. To make the most use out of this information, users need to share the information with one another.

Existing research in anonymized data sharing impose significant computational and/or bandwidth requirements making them difficult to deploy in practical scenarios. In particular [12] proposes an anonymous, gossip-like protocol meant for aggregating data from smart devices. This support for generality in data queries is advantageous, but the protocol requires several rounds of inter-node communications and heavy use of encryption to protect against fraudulent transactions. NILM-Net, specifically designed for electric load data, leverages the physical constraints of the power grid to detect and eliminate fraud with minimal network communication and encryption.

III. NILM-Net

NILM-Net collects load event data produced by independently operated power monitors and creates an anonymized energy database for use by grid participants. Each power monitor measures a single building such as a home or business and identifies load events in the aggregate power waveform using NILM techniques discussed previously. Each event is encapsulated in a packet with a timestamp, power information, and unique identifier (UUID) as shown in Figure 3. The UUID is a 128-bit value randomly generated by the node [13]. Since there is no central coordination, the very large bit-width ensures nodes do not coincidentally generate the same value. The metadata is an optional field that provides additional information such as load type, time sensitivity (critical, flexible schedule, periodic, etc.). Metadata improves the utility of the data but does so at the expense of user privacy. A monetary compensation scheme discussed in Section III-B enables nodes to capture this additional value and therefore decide their own balance of this privacy trade-off. Finally, the duplication factor and salt field ensure the fair operation of the network and are discussed in detail later in this section.

A. Privacy

Nodes anonymize their power usage by swapping data packets before submitting them to the central server in a process called gossip (Figure 4). Packets must be submitted to the server within 24 hours. Until then they may be exchanged asynchronously and arbitrarily between nodes. There is no limit to the number of packets in an exchange or the number of times an individual packet may be exchanged. The data remains anonymous to both the nodes and the server as long as there are at least two exchanges. Under this assumption, a node receiving transients from a peer is unable to distinguish between packets originating from that peer and packets forwarded by that peer from an earlier exchange. When the server ultimately receives the data it too is unsure of the data source which guarantees the anonymity of the database records.

Gossip protects user privacy but introduces new problems by relying on intermediaries to forward information. In particular, packets may not reach the destination if an intermediate node goes offline before submitting it to the server. To avoid dropped packets nodes may duplicate information by sending

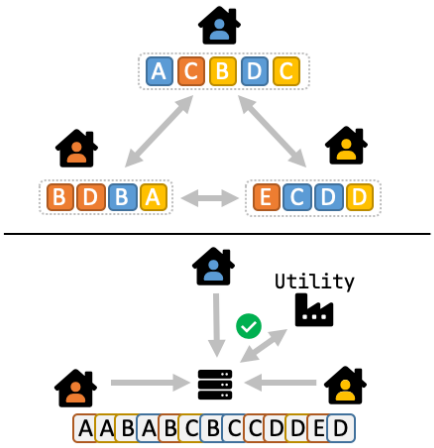

Fig. 4. The NILM-Net gossip protocol. Nodes itemize power transients into data packets. Any time before a packet expires nodes may choose to swap it with a peer (top) or submit it to the server (bottom). Swapped packets may be original or received from an earlier exchange, and may also be duplicated. The server can verify received transients are physically plausible based on aggregate utility measurements.

the same packet to multiple peers. The unique packet ID ensures the server only stores a single copy of each load event in the database even if multiple copies are submitted.

B. Compensation

To encourage participation, NILM-Net incorporates a monetary compensation mechanism to reward users that submit useful data. Database queries are charged a variable rate depending on the quantity and granularity of the data requested. Queries which request metadata fields such as load type are charged higher rates than those which only request power data. Query funds are distributed evenly among the nodes that submitted the records used in the response. When a node duplicates a packet it multiplies the `df` field by the number of new packets. For example a node that wants to send the same load event to two different peers would set `df=2`. If one of these nodes then sends the data to three peers, it exchanges packets with `df=6`. The server uses the reciprocal of `df` to determine the probability a node is awarded ownership of that data. Therefore packets with a higher `df` have a lower relative value since they are less likely to provide credit to the submitter. To deter nodes from manipulating `df` in hopes of securing more valuable packets in a gossip exchange the server verifies that the sum of the reciprocal `df` values is ≤ 1. If this constraint is violated the server conducts a fraud investigation to determine the culprit node(s), a process discussed in detail in Section IV. As long as the cost function used by the server is well known and the `df` values are accurate every node should agree on the value of packets exchanged in gossip and profit-maximizing participants will therefore choose to only exchange similarly valued sets of packets. As long as packet exchanges are equitable, nodes will receive an average compensation proportional to the value of data they produce even though they never receive credit for their own data.

C. Fraud Detection

Gossip networks of any practical size and especially ones with monetary incentives must expect some level of misbehavior by nodes seeking to disrupt the proper flow of information for personal gain or simply to exploit the system. NILM-Net provides strong protections against such data fraud. When malicious data modification occurs participants can collaboratively reverse the gossip sequence to reveal the identity of the culprit node(s) without sacrificing the anonymity of the data source. The fraud detection mechanics are shown in Figure 5. The Hash Registry is a trusted authority that stores timestamped hashes submitted by nodes. In addition to the Hash Registry, NILM-Net also maintains a Public Key Infrastructure (PKI) which allows nodes to digitally sign messages. When a node forwards a packet during gossip it provides a signed receipt and an index to the hash of this receipt in the Registry. The sender (`TX`) produces the receipt by signing the packet concatenated with the recipient (`RX`) node's ID (such as `nodeXX.nilm.net`) as shown below:

$$\text{receipt} = \text{SIG}_{\text{TX}}(\text{packet} || \text{ID}_{\text{RX}}) \quad (1)$$

The recipient can verify the contents of the receipt using the sender's public key and can also check that the correct hash is available in the Registry. If the packet happens to be fraudulent the recipient can use the receipt to absolve itself and blame the sender. It is important to note that the Registry stores the hash of the receipt and not the receipt itself. Hashes uniquely identify data but do not provide any information about the data itself. This makes registry entries a type of cryptographic commitment, allowing nodes to selectively reveal portions of the gossip sequence.

Every 24 hours the server publishes a list containing the hashes and IDs of all packets it has received over this period. This allows the data author as well as any intermediate nodes in the gossip sequence to verify whether a packet was received correctly. If the published hash does not match a node's local version of that packet (based on the ID) it indicates the data was fraudulently changed in transit. While hashes are irreversible it may be possible for a motivated attacker to construct potential data packets and test whether the hash is in the published list. This is possible because the space of valid packets is relatively small. To prevent this type of reverse lookup, every packet has an additional salt field (Figure 3). This random data greatly increases the space of the input and makes it computationally infeasible for an attacker to infer any packet data from the published hashes. If a node finds an incorrect hash, it submits a fraud claim to the server. A fraud claim consists of three parts: the correct packet, a receipt of this packet, and a Hash Registry index to this receipt. The server uses this information to conduct a fraud investigation as discussed in the next section.

IV. Threat Scenarios

In order to provide reliable value to the smart grid, NILM-Net must detect malicious actors trying to circumvent the protocol. This section explains the fraud detection mechanisms

that ensure the fair operation of the system. When fraud does occur, these mechanisms can accurately attribute blame while maintaining user privacy. Several threat scenarios are discussed which, while not exhaustive, serve to illustrate the operation of the network under attack. For simplicity, scenarios have the minimum number of nodes needed to demonstrate the attack but these fraud controls are believed to be sufficient to protect against arbitrarily complex data manipulation schemes.

There are two general types of fraudulent behavior on the network: data injection and data manipulation. Data injection attacks are spurious packets that do not correspond to actual power events, and data manipulation attacks are modifications made to transients during the gossip exchange. Data injection threats are limited by the fact that the power grid is a closed system. Nodes which inject false transients can be detected by comparing their submitted transients to the output of the power producers over the same time period. If a majority of consumers in a grid participate in NILM-Net, the power produced should roughly match the sum of the itemized transients submitted to the server. In large grids where NILM-Net participation may be low, injection attacks can still be detected by comparison with AMI data. Care must be taken to properly anonymize the AMI data such as implementing an oracle as shown in Figure 4 rather than providing direct access to the meter data.

A. Fraudulent Modification

Data modification attacks, such as altering packet metadata, are a more sophisticated threat vector. Malicious nodes may modify data to increase the apparent value of a packet in an exchange (by adding additional or highly specific metadata) or to tamper with the accuracy of the load event database. Modification attacks are possible because transient data is sent as plain text between nodes. Plain text allows nodes to estimate the value of packets received in a swap and reduces the computational overhead of the protocol. If the server is alerted to a fraudulent packet, either because it is not physically plausible or because a participant has flagged it as corrupt by finding a discrepancy in the published packet hashes, the server determines which node to blame. This is inherently a complex problem. The node that delivered the data will insist (whether true or false) that it simply received the corrupted data from someone else – indeed the entire protocol is designed to make it difficult to identify the data producer. Figure 5 illustrates the network operation when a node maliciously modifies a packet in transit. Data is generated by Node A which exchanges the packet and associated receipt with Node B. Node B chooses to continue the gossip sequence by forwarding the packet to C but modifies the data in some way first. In order for C to accept the packet, B must produce a valid receipt for the modified packet and also submit the matching hash to the Registry. This will eventually implicate B when the fraud is discovered.

The fraud is revealed when the server publishes the hash list and A realizes its packet was altered. The fraud investigation sequence is shown in the lower half of Figure 5. Node A initiates the process by submitting a fraud claim consisting

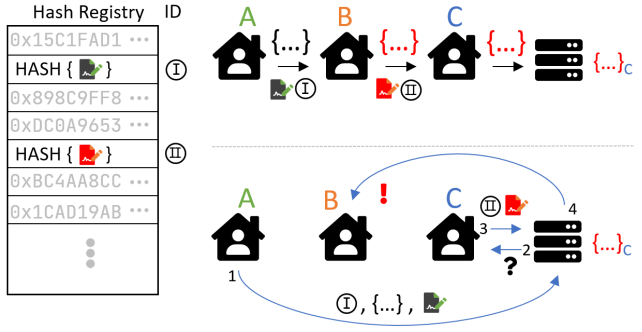

Fig. 5. Detecting fraudulent data modification. Node B illegally modifies data received from A (top). Sometime later, Node A alerts the server to the discrepancy (1). The server first accuses Node C (2) which provides receipt II to prove its innocence (3). The server then accuses B which cannot provide a satisfactory response and is correctly charged with fraud.

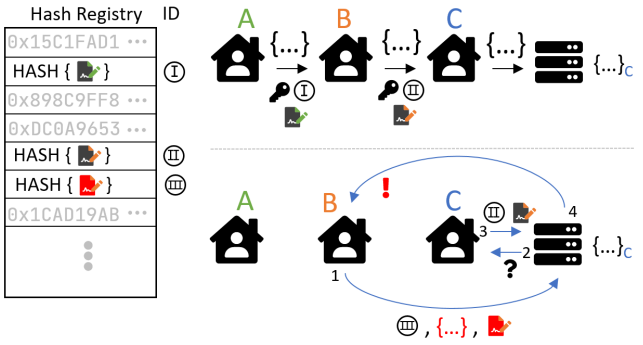

Fig. 6. Detecting fraudulent accusations. Node B correctly forwards the packet from A (top) but later fabricates a fraud claim hoping to implicate C (bottom, 1). The claim is handled in the same manner as Figure 5 with the server charging C (2) and C responding with exonerating evidence (3). Registry entry II conflicts with III correctly implicating Node B as the culprit.

of the original packet, the receipt showing transmission to B and the Registry index of this receipt. The server verifies the claim by confirming the receipt matches the packet contents and the corresponding Registry entry occurs earlier than than the submission of the packet from Node C. Since packet IDs are unique this proves that some intermediate node modified the packet between Node A and the server. The server assumes the submitter, Node C, is guilty until proven innocent. Node C exonerates itself by revealing the receipt from B and the corresponding Registry entry, II, which occurs after I but before the submission of the packet to the server. The server then charges node B who cannot produce a similar set of exonerating data and is therefore guilty of fraud. In this scenario A is both the data producer and the fraud claimant but any node in a gossip sequence can submit a fraud claim if its local version of the packet does not match the published hash. Thus, A's identity is still protected because the server cannot distinguish between a claim made by the data author or an intermediate node in a longer gossip sequence.

B. Fraudulent Accusation

Rather than directly modifying packets, an attacker may sow distrust among participants by generating spurious fraud

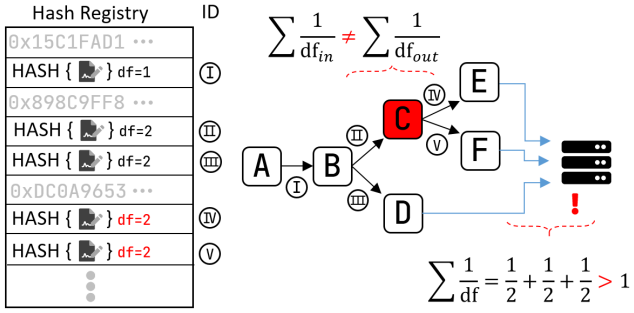

Fig. 7. Fraudulent duplication occurs when a node replicates packets without updating the duplication factor. The server can detect this attack by computing the sum of the reciprocal `df` values and determine the fraudulent node by examining the gossip receipts.

claims hoping to implicate innocent nodes. This type of attack is shown in Figure 6. Here Node `B` correctly forwards the packet from `A` and the unmodified data arrives at the server. Later `B` submits a fraud claim by creating a modified packet and submitting the corresponding hashed receipt to the Registry. From the server's perspective this claim looks identical to the claim in Figure 5. Initially `C` looks guilty because as before the claim documents are valid: The packet IDs match with different content and the receipt is registered with a timestamp prior to the packet submission (entry III). However `C` can reveal receipt II proving it received the unmodified data from `B`. Node `B` is therefore exposed as the culprit because it registered two receipts for the same packet with different data (the relative time ordering of II and III is not important).

C. Fraudulent Duplication

When the server receives multiple copies of the same packet from different nodes it randomly assigns the query reward to only one node based on the duplication factor. To maximize expected revenue a malicious node may covertly duplicate a packet without modifying `df`, as shown in Figure 7. The server can detect this type of attack and identify the malicious node(s) in a similar manner to the data modification attack discussed previously. The server is initially alerted to the attack by computing an invalid sum of duplication factors. At this point the server accuses all of the submitters of fraud but `E`, `F`, and `D` are all exonerated by revealing receipts (IV, V, and III respectively) which show that they are accurately forwarding data received from other nodes. This implicates both `B` and `C`. Node `B` reveals receipt I which has a lower `df` than any copies of the packet it has forwarded (at this point only III has been revealed). With both III and V revealed, Node `C` must reveal an inbound receipt with `df`=1 which it cannot since receipt II is for a packet with `df`=2.

V. Conclusion

A practical implementation of NILM-Net will include many challenges. Although NILM has made great strides in load identification transients may be unidentifiable or misclassified. A practical system needs to be robust to these errors, when collating the data, issuing monetary rewards, and penalizing nodes. In addition, a practical implementation would require protocol specifications for node entry and exit from the energy network. Users may want to create a "blacklist" of power transients blocked from being sent to the central server to keep certain activities private.

These challenges represent exciting areas of future research. NILM is a promising technology but without a framework for data sharing, its utility is limited. A centralized energy database can provide value across the smart grid. NILM technology makes this possible with a minimal amount of additional infrastructure. However, a energy database must respect user privacy. Load event data is a valuable commodity and energy consumers should be compensated for sharing this information. Not only is this ethically preferable, it also incentives participation making it more likely such as system while gain wide adoption. NILM-Net provides a solution that scales, gives users control over sharing their data, and rewards them for doing so.

References

[1] I. Zaman and M. He, "A multilayered semi-permissioned blockchain based platform for peer to peer energy trading," in *2021 IEEE Green Technologies Conference (GreenTech)*, 2021, pp. 279–285.

[2] Y. Seyedi, H. Karimi, and S. Grijalva, "Irregularity detection in output power of distributed energy resources using pmu data analytics in smart grids," *IEEE Trans. Ind. Inf.*, vol. 15, no. 4, pp. 2222–2232, 2019.

[3] M. R. Abid, A. Khallaayoun, H. Harroud, R. Lghoul, M. Boulmalf, and D. Benhaddou, "A wireless mesh architecture for the advanced metering infrastructure in residential smart grids," in *2013 IEEE Green Technologies Conference (GreenTech)*, 2013, pp. 338–344.

[4] M. H. Saeed, W. Fangzong, B. A. Kalwar, and S. Iqbal, "A review on microgrids' challenges & perspectives," *IEEE Access*, vol. 9, pp. 166 502–166 517, 2021.

[5] H. V. Dang, M. Tatipamula, and H. X. Nguyen, "Cloud-based digital twinning for structural health monitoring using deep learning," *IEEE Trans. Ind. Inf.*, vol. 18, no. 6, pp. 3820–3830, 2022.

[6] A. Saad, S. Faddel, T. Youssef, and O. A. Mohammed, "On the implementation of IoT-based digital twin for networked microgrids resiliency against cyber attacks," *IEEE Trans. Smart Grid*, vol. 11, no. 6, pp. 5138–5150, 2020.

[7] K. Carrie Armel, A. Gupta, G. Shrimali, and A. Albert, "Is disaggregation the holy grail of energy efficiency? the case of electricity," *Energy Policy*, vol. 52, pp. 213–234, 2013, special Section: Transition Pathways to a Low Carbon Economy.

[8] R. Nieto, L. de Diego-Otón, Á. Hernández, and J. Ureña, "Data collection and cloud processing architecture applied to nilm techniques for independent living," in *2021 IEEE I2MTC*, 2021, pp. 1–6.

[9] Y.-C. Chen, C.-M. Chu, S.-L. Tsao, and T.-C. Tsai, "Detecting users' behaviors based on nonintrusive load monitoring technologies," in *2013 10th IEEE ICNSC*, 2013, pp. 804–809.

[10] R. Rashed Mohassel, A. Fung, F. Mohammadi, and K. Raahemifar, "A survey on advanced metering infrastructure," *International Journal of Electrical Power & Energy Systems*, vol. 63, pp. 473–484, 2014.

[11] G. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, 1992.

[12] E. Tremel, K. Birman, R. Kleinberg, and M. Jelasity, "Anonymous, fault-tolerant distributed queries for smart devices," *ACM Trans. Cyber-Phys. Syst.*, vol. 3, no. 2, Oct 2018.

[13] R. S. P. Leach, M. Mealling, "A Universally Unique IDentifier URN Namespace," Internet Requests for Comments, RFC Editor, RFC 4122, July 2005.