AMIDS: A Multi-Sensor Energy Theft Detection Framework for Advanced Metering Infrastructures

Stephen McLaughlin and Brett Holbert
Computer Science and Engineering
Pennsylvania State University
{*smclaugh, bdh5027*}*@cse.psu.edu*

Saman Zonouz
Electrical and Computer Engineering
University of Miami
s.zonouz@miami.edu

Robin Berthier
Information Trust Institute
University of Illinois
rgb@illinois.edu

***Abstract*—The advanced metering infrastructure (AMI) is a crucial component of the smart grid, replacing traditional analog devices with computerized smart meters. Smart meters have not only allowed for efficient management of many end-users, but also have made AMI an attractive target for remote exploits and local physical tampering with the end goal of stealing energy. While smart meters posses multiple sensors and data sources that can indicate energy theft, in practice, the individual methods exhibit many false positives. In this paper, we present AMIDS, an AMI intrusion detection system that uses information fusion to combine the sensors and consumption data from a smart meter to more accurately detect energy theft. AMIDS combines meter audit logs of physical and cyber events with consumption data to more accurately model and detect theft-related behavior. Our experimental results on normal and anomalous load profiles show that AMIDS can identify energy theft efforts with high accuracy. Furthermore, AMIDS correctly identified legitimate load profile changes that more elementary analyses classified as malicious.**

I. Introduction

The Advanced Metering Infrastructure (AMI) is changing the way electricity is measured, consumed, and even distributed. Digital smart meters remotely report not only fine-grained energy consumption data, but also logs of events indicating malfunctions, misconfigurations, and potential physical tampering. These monitoring capabilities, coupled with large-scale AMI data aggregation promise to significantly mitigate the problem of energy theft, an especially pervasive problem in developing countries.

However, the recent nation-wide AMI deployment effort has had quite an opposite effect by fueling concerns about new ways to steal power, e.g., through remote smart meter compromise. For instance, in 2009, the FBI reported a wide and organized energy theft attempt that may have cost up to 400 million dollars annually to a utility following an AMI deployment [1]. Indeed, AMI significantly increases the attack surface that utilities have to protect by introducing new cyber threats on physically-accessible devices [18]. Penetration testing efforts have shown vulnerabilities in smart meters that could lead to stealthy energy fraud. Additionally, remote meter reading eliminates the monthly visit by technicians to record consumptions and to visually inspect meters.

As a result, the need for an efficient monitoring solution to detect energy theft attempts in AMI has never been more critical. In this paper, we introduce AMIDS, an integrated cyber-physical intrusion detection system to identify malicious energy theft attempts. AMIDS differs from previous solutions by evaluating multiple AMI data sources under a combination of techniques to detect theft-related behavior while reducing false positives. In particular, AMIDS uses an attack graph based information fusion technique to conceptually combine collected evidences from three types of AMI-specific information sources: 1) cyber-side network- and host-based intrusion detection systems; 2) on-meter anti-tampering sensors; and 3) power measurement-based anomalous consumption detectors through nonintrusive load monitoring (NILM). The main contributions of this paper are as follows:

- We present an information fusion solution which makes use of an AMI-specific attack graph to identify energy theft attempts with minimum number of false positives.
- We leverage data mining techniques to identify energy theft through nonintrusive load monitoring. We designed two algorithms: a supervised approach that can identify individual appliance consumption and an unsupervised approach that learns by clustering load events.
- We build a realistic household load simulator that we used to evaluate the different individual detection techniques and the information fusion solution through the injection of realistic energy theft attacks.

This material is based upon work supported by the Department of Energy under Award Number DE-OE0000097.

II. Related Work

Several solutions to energy theft have been proposed recently [9]. A popular approach has been to apply support-vector machine (SVM) to energy consumption profiles [8], [20]. This approach consists of training an SVM from a historical dataset and then testing the SVM on a different dataset to find anomalies in the customer energy consumption. [8] reports an accuracy of 98.4% based on a training set of 440 instances and a testing set of 220 customers. The same authors extended their approach in [10] to leverage a hybrid neural-network model and encoding technique in order to automatically set the many parameters required by the model.

A different approach is shown in [4]. Here, the focus is on identifying problematic metering installations, e.g., due to malfunction or energy theft, through a central observer meter in each neighborhood. Neighborhood energy use is compared with individual customer loads using a model of N linearly independent equations. This model is solved using matrix inversion and recursive statistical methods, i.e., least squares. This approach is limited by its reliance on linear independence of equations and zero resistance of power lines.

A radically different approach is taken in [7] by using a harmonic generator to actively deteriorate appliances of customers who steal energy. Sensors monitor consumption values, identify suspicious non-technical losses, disconnect genuine customers, operate the harmonic generator for few seconds, and reconnect everyone. An important limitation of

this solution is that if on-meter harmonic sensors fail, damage to genuine customers could make the cost of false positives prohibitively high.

III. Threat Model

There are a variety of known techniques for energy theft that we assume an adversary may attempt against an AMIDS-equipped AMI deployment. At the level of customer homes, the most common techniques involve either tapping an external source such as a neighbor or distribution feeder, or meter tampering to inhibit proper recording of consumption. The latter may be done by applying magnets to interfere with instruments such as electromechanical rotors or solid state current transformers, or by reversing or disconnecting meters from their sockets. At the grid level, energy thieves usually bypass meters by wiring power hungry appliances directly to the grid, or connecting their entire electric system to a feeder with a pirate transformer.

Addressing such physical tampering-related issues has been one of the benefits of AMI. Indeed, smart meters can detect and report certain tampering attempts, and solid-state metering mitigates some attack techniques targeting electromechanical meters. But, the addition of network communication and smart devices to the grid has also brought new attack vectors. For example, it is trivial for a customer to jam meter wireless communications to suppress physical tampering alarms. Cyber attack techniques against AMI have been recently studied [16], [17] and include interrupting measurements, gaining privileged access to the meter firmware, tampering with the meter storage, and intercepting the meter communications to block or alter consumption values being reported.

To summarize, we classify energy theft techniques into three categories: 1) physical attacks, 2) cyber attacks, and 3) data attacks having an impact on power measurements. Note that attacks in the third categories are made possible through attacks from the first and second categories. The different attacks are detailed in Table I. The table is used in the following sections as a guide to ensure a comprehensive coverage of the threats from the described detection solutions. Moreover, we draw from these attack techniques to simulate attack scenarios in order to evaluate AMIDS.

TABLE I
Mapping between attacks and detection techniques

Id	*Attack technique*
	Cyber
A_{c1}	Compromise meters through remote network exploit
A_{c2}	Modify the firmware/storage on meters
A_{c3}	Steal credentials to login to meters
A_{c4}	Exhaust CPU/memory
A_{c5}	Intercept/alter communications
A_{c6}	Flood the NAN bandwidth
	Physical
A_{p1}	Break into the meter
A_{p2}	Reverse the meter
A_{p3}	Disconnect the meter
A_{p4}	Physically extract the password
A_{p5}	Abuse optical port to gain access to meters
A_{p6}	Bypass meters to remove loads from measurement
	Effect on power measurements
A_{d1}	Stop reporting entire consumption
A_{d2}	Remove large applicances from measurement
A_{d3}	Cut the report by a given percentage
A_{d4}	Alter appliance load profile to hide large loads
A_{d5}	Report zero consumption
A_{d6}	Report negative consumption (act as a generator)

IV. Individual Energy Theft Detection Mechanisms

A. Physical Tampering Detection Solutions

Smart meters are already equipped with sensors to collect and log potential physical tampering events such as removal of the meter cover and physical bumping of the meter. However, a problem with some such alerts is the high rate of false positives. For example, a heavy truck passing near a meter can trigger the tilt alert [15]. Thus, we include such tamper detection sensors in our solution to detect physical attacks, but false positive are reduced by combining tampering alerts with additional data sources covered in the following two sections. From our threat models described in Table I, meter tamper alerts provide the following observations:

- *Observation* O_8: anti-tampering alert to detect A_{p1}
- *Observation* O_9: reverse rotation alert to detect A_{p2}
- *Observation* O_{10}: disconnect alert to detect A_{p3}
- *Observation* O_{11}: anti-tampering alert to detect A_{p4}

B. Cyber Intrusion Detection Systems

To address the challenge of detecting cyber attacks introduced by AMI, AMIDS leverages two complementary intrusion detection systems that can be implemented and deployed via firmware upgrade: 1) a remote cumulative attestation kernel (CAK) in meters [13], and 2) a specification-based network intrusion detection systems deployed on access points or dedicated sensors in the local neighborhood area network [5]. A CAK is a lightweight solution for embedded systems such as meters, which records an unbroken sequence of application firmware upgrades. This audit log can be remotely queried by a verifier to detect firmware tampering, e.g., due to remote exploitation. The specification-based network intrusion detection system monitors traffic among meters and access points across layers to ensure that devices are running in a secure state and their operations respect a specified security policy. It does this by constraining communications made using the ANSI C12.22 standard protocol, thus guaranteeing that all policy violations will be detected. The soundness of these constraints can be formally verified.

Our cyber intrusion detection system provides the following observation capabilities to cover attacks from our threat models described in Table I:

- *Observation* O_1: spec.-based network monitoring to detect A_{c1}
- *Observation* O_2: remote firmware attestation to detect A_{c2}
- *Observation* O_3: spec.-based monitoring and meter authentication logs to detect A_{c3}
- *Observation* O_4: spec.-based monitoring and meter responsiveness to detect A_{c4}
- *Observation* O_5: spec.-based monitoring to detect A_{c5}
- *Observation* O_6: spec.-based monitoring to detect A_{c6}
- *Observation* O_7: remote firmware attestation to detect A_{p5}

C. Power Measurement-based Anomaly Detection

The third class of observations to detect theft-related behavior leverages the fine-grained load profile data available from smart meters. In particular, individual load profile events are analyzed to identify appliances being turned on and off. The results are used to create a usage *profile* for each household. These profiles will be used later to detect changes in household energy consumption patterns. In particular, we introduce

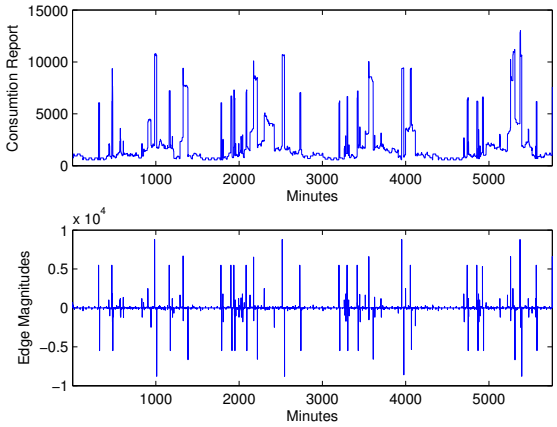

Fig. 1. A Sample 4-Day Load Profile and Identified Edges

two power measurement-based detection solutions based on supervised and unsupervised machine learning techniques. The algorithms are based on Naive Bayes learning that employs the method of maximum likelihood and is known to be one of the most effective and efficient classification algorithms in complex real-world situations.

We review the Naive Bayes algorithm briefly, and then discuss our two load-based energy theft detection solutions. Formally, the probability model for a classifier is a conditional model $Pr(C|F_1,F_2,\cdots,F_n)$ over a dependent class variable C that takes on a binary value, 0 (legitimate power consumer) and 1 (anomalous power measurements). F_i represents the i-th feature. Using a Bayes' theorem,

$$Pr(C|F_1,F_2,\cdots,F_n) = \frac{Pr(C)\cdot Pr(F_1,F_2,\cdots,F_n|C)}{P(F_1,F_2,\cdots,F_n)} \quad (1)$$

can be derived, and given the independence assumption,

$$Pr(C|F_1,F_2,\cdots,F_n) = \frac{1}{Z}P(C)\cdot\prod_{i=1}^{n}Pr(F_i|C), \quad (2)$$

where Z is a constant scaling factor representing the evidence. Given the above probability model, the Bayesian classifier combines this model with a decision rule. In particular, the hypothesis with the maximum a posteriori is picked:

$$C(f_1,f_2,\cdots,f_n) = \arg\max_{c\in\{0,1\}} P(C=c)\cdot\prod_{i=1}^{n}P(F_i=f_i|C=c). \quad (3)$$

Supervised Anomaly Detection. The supervised technique labels each on or off edge in the load profile according to its appliance of origin. The algorithm then determines which appliances $a \in A$ are missing from power measurements, i.e., if the mode of theft bypassed some appliances around the meter. The algorithm has two learning phases. First, a database of appliance signatures is created and stored for use by a Non-Intrusive Load Monitor (NILM). The NILM uses this database to identify appliance usage in the home over time. Second, AMIDS learns the daily usage frequencies of each individual appliance using appliance data provided by the NILM. More specifically, the power consumption time series are analyzed and the (*edges*) $E = (e_{t_0}, e_{t_1}, \cdots, e_{t_n})$ corresponding to on/off events are identified and recorded. Each edge magnitude represents one or more appliance events. Figure 1 shows 1) a sample power consumption time series of a single household generated by our implementation that simulated turn on/off

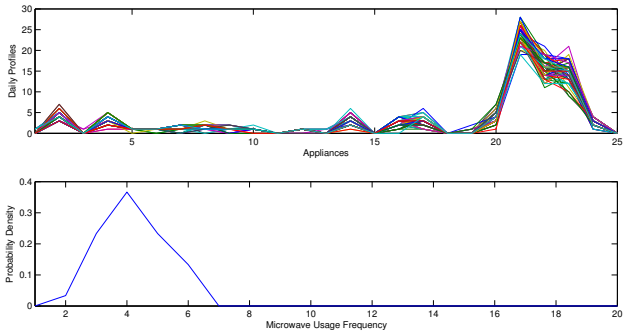

Fig. 2. Learned Normal Appliance Usage Profiles

incidents of 25 different home appliances, and 2) the identified edges within the same trace. The NILM works by solving the following binary integer programming problem to determine which devices contributed to a given edge.

$$\begin{array}{ll} \min & B^T x \\ \text{s.t.} & Qx \le e_{t_i} + \delta \\ & -Qx \le -e_{t_i} + \delta \\ & x \ge 0 \end{array} \quad (4)$$

where $B = [1,1,\cdots,1]_{2\cdot|A|\times 1}$; $Q = [Q_p; -Q_p]$, in which Q_p is an $|A|$-dimensional vector of power appliance consumption profiles, and $[a;b]$ represents the concatenation of the vectors a and b. This integer programming problem is solved to get the $2\cdot|A|$-dimensional binary vector x, where an element represents whether its corresponding appliance contributed to the edge e_{t_i}. Here, δ is a small threshold value to account for measurement noise. The objective of the optimization is to minimize number of incidents per edge. This is a reasonable assumption as many near-simultaneous appliance events are unlikely [12].

Once the set of appliances contribution to each edge is identified, AMIDS learns based on the daily frequency of each appliance f_a. Thus, over an n-day learning phase, a usage profile matrix

$$U_{h_i} = \begin{pmatrix} f_{a_1,d_1} & f_{a_2,d_1} & \cdots & f_{a_{|A|},d_1} \\ f_{a_1,d_2} & f_{a_2,d_2} & \cdots & f_{a_{|A|},d_2} \\ \vdots & \vdots & \ddots & \vdots \\ f_{a_1,d_n} & f_{a_2,d_n} & \cdots & f_{a_{|A|},d_n} \end{pmatrix} \quad (5)$$

per household h_i is saved. Each column is then used to calculate the probability mass distribution $P_{h_i,a_j}(v)$ $v \in \mathbb{Z}$ that appliance a_j is used v times per day in household h_i. This completes the profiling phase. Figure 3 shows our implementation results: 1) home appliance usage frequency reports of a single household over 20 days (each line represents a single day); and 2) the empirical probability mass distribution of the microwave usage frequency per day.

The calculated profiles (distributions) are used for anomaly detection purposes with the Bayesian classifier. The objective is to mark a given day-long smart meter measurements as normal or anomalous based on that household's profile. In particular, the prior class probability $P(C)$ in Equation (3) can be obtained from existing energy theft data [2], and the conditional distributions are obtained from the learned profiles. Here, a features F_i is the daily usage frequency of appliance i. Figure 3 shows our evaluation results for the supervised detection of anomalous power measurements. In

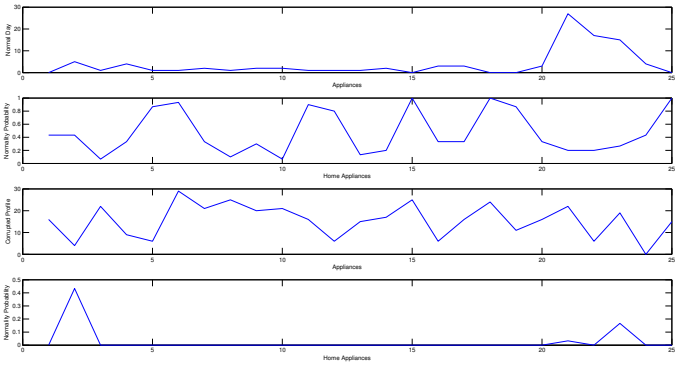

Fig. 3. Normal Day Profile and Classification Probability

particular, the first and second graphs show a normal trace for a single household over a day and the posterior distribution for individual appliances. As shown in the third and fourth graphs, a corrupted measurement trace leads to a significant reduction in the posterior distribution values (indicating that the reported measurements are less likely to be normal).

We note that the use of NILMs along side smart meters has raised privacy concerns [21]. Recent studies have shown that NILMs can reveal home occupant behaviors [14], [19]. While we defer the design of privacy-preserving protocols [23] for our scheme to future work, we mention a practical measure to mitigate leaks of most legitimate user's consumption patterns. Fine grained data for usage by load-based detection schemes can be released only after physical or cyber tampering alarms have been raised.

Unsupervised Anomaly Detection. The unsupervised detector groups individual load events into clusters based on their real-power magnitude. Thus, appliances with similar load sizes will be placed in the same cluster. The resulting individual clusters are more sensitive to load changes than the net load. For example, bypassing a single appliance will have a noticeable effect on the cluster containing the appliance, even if the change in net load is very small. The unsupervised learning algorithm proceeds as follows. Edge detection is first used to extract a set of events $f_1, f_2, \ldots, f_n$ (positive or negative edges) from the load profile. K-means clustering is then done based on individual event magnitudes, resulting in a set of clusters $\mathcal{C}$ with $c = \{f_1, f_2, ..., f_{|c|}\}$ for all clusters $c \in \mathcal{C}$. The number of clusters $|C|$ is determined by maximizing the average *silhouette* value s across all clusters.

$$s = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \frac{1}{|c|} \sum_{f \in c} \frac{b(f) - a(f)}{max\{b(f), a(f)\}} \tag{6}$$

Here, $b(f)$ is the Euclidean distance between f and all events in other clusters, and $a(f)$ is the distance between f and all events in its own cluster. Given an optimal clustering, the upper and lower bounds on each cluster are found and used to bucket events during normal operation. Bayesian classification is then done over the distribution of bucketed data against the clustering of the training data.

An example clustering of three datasets is shown in Figure 4 with four clusters formed from each dataset. The three datasets are as follows. (1) The solid line shows the probability density function (pdf) of events per day in each of four clusters from training data. (2) The dashed line is the pdf of events in a clustering of the same scenario with an HVAC system that is 30% more efficient than the baseline. (3) The line with $\times$ marks has the HVAC bypassing the meter. As can be seen, the clustering of the malicious test case differs significantly from the baseline and legitimate test cases.

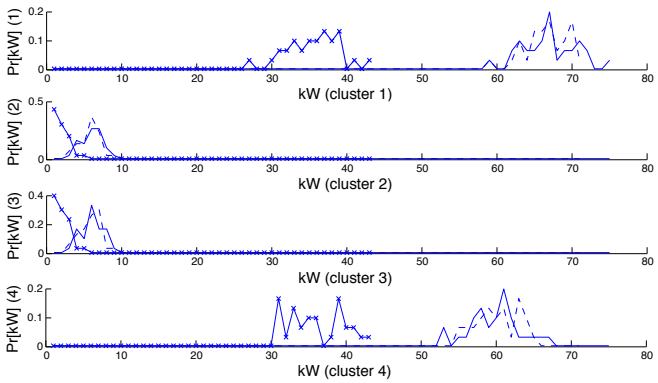

Fig. 4. Unsupervised Learning of Basline (solid), Legitimate (dashed), and Malicious $\times$ Profiles.

In summary, the power-measurement monitoring system provides the following observation capabilities to cover attacks from our threat models described in Table I:

- *Observation* O_{12}: supervised and unsupervised anomaly systems to detect A_{p6}
- *Observation* O_{13}: utility-side report freq. checkers to detect A_{d1}
- *Observation* O_{14}: supervised anomaly system to detect A_{d2}
- *Observation* O_{15}: aggregated monthly changes to detect A_{d3}
- *Observation* O_{16}: supervised anomaly system to detect A_{d4}
- *Observation* O_{17}: unsupervised anomaly system to detect A_{d5}
- *Observation* O_{18}: utility-side negative consumption alert to detect A_{d6}

V. Multi-Source Information Fusion

Alerts from each of the security sensors discussed in Section IV indicate individual attack steps against AMI. However, as proved in practice, these sensors report fairly large numbers of false positives and sometimes miss intrusions; therefore, reporting energy theft solely based on individual alerts will result in many costly physical inspections. To improve the overall accuracy, AMIDS makes use of a novel model-based solution to correlate alerts and provides operators with contextual information. In particular, AMIDS leverages a set of common energy theft attack paths, i.e., the different ways that an energy theft attack could occur, to reduce false positives due to individual false alarms.

AMIDS uses an attack graph-based information fusion algorithm to combine evidence of on-going attacks from multiple sources. Figure 5 shows a simplified energy theft attack graph for a smart meter. The attack graph is a state-based directed graph which models various attack paths starting from the initial state s_0 and continues until the goal of theft (state s_g) is reached. At each node, the security state of the smart meter is identified by the following two binary values. 1) The attacker's current *privilege* in the meter: this captures what the attacker can do in the future, and is either none $\varnothing$ or the administrator privilege M. 2) The security *consequences* of attacker actions: this captures the set of actions the attacker has accomplished such as a modified meter firmware or exhausted CPU on the meter.

As shown in the figure, there are specific alerts and intrusion detection methods to identify each malicious action needed to proceed through the graph. Because these individual alerts are subject to false positives, AMIDS makes use of the

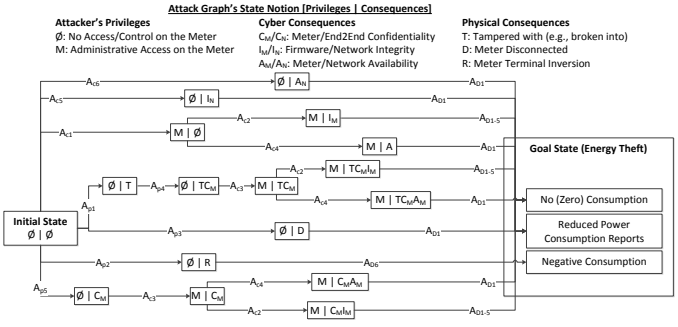

Fig. 5. A Simplified Cyber-Physical Attack Graph for AMI

attack graph to detect energy theft efforts by correlating alert sequences denoting a complete energy theft attack, i.e., a path from s_0 to s_g.

To perform information fusion online, AMIDS considers the attack graph as a hidden Markov model (HMM) [22] and the alerts triggered by different detection techniques as observables $o_i \in \mathbf{O}$. Formally, AMIDS considers each attack path as a discrete-time hidden Markov process, i.e., event sequence $Y = (y_0, y_1, \cdots, y_{n-1})$ of arbitrary lengths. $y_i = (s_i, o_i)$, where s_i is an HMM state at the ith step of the attack and is unobserved, and the observation o_i is the set of triggered intrusion detection alerts at that step. AMIDS's main responsibility is to compute $Pr(s_t \mid o_{0:t})$, that is, the probability distribution over hidden states at each time instant, given the HMM model and the past IDS alerts $o_{0:t} = (o_0, \cdots, o_t)$. In particular, AMIDS makes use of the forward-backward smoothing algorithm [22], which, in the first pass, calculates the probability of ending up in any particular HMM state given the first k alerts in the sequence $Pr(s_k \mid o_{0:k})$. In the second pass, the algorithm computes a set of backward probabilities that provide the probability of receiving the remaining observations given any starting point k, i.e., $Pr(o_{k+1:t} \mid s_k)$. The two probability distributions can then be combined to obtain the distribution over states at any specific point in time given the entire observation sequence:

$$Pr(s_t \mid o_{0:t}) = Pr(s_k \mid o_{1:k}, o_{k+1:t}) \propto Pr(o_{k+1:t} \mid s_k) \cdot Pr(s_k \mid o_{1:k}), \quad (7)$$

where the last step follows from an application of Bayes's rule and the conditional independence of $o_{k+1:t}$ and $o_{1:k}$ given s_k. Having solved the HMM's smoothing problem for $Pr(s_t \mid o_{0:t})$, AMIDS probabilistically knows about the current state. Consequently, AMIDS picks the state with highest probability using the Most Likely State (MLS) technique [6] $s^* = \arg\max_s Pr(s_t \mid o_{0:t})$ and triggers the energy theft alert if $s^* = s_g$.

VI. Experimental Evaluations

A. Load Profile Datasets

1) Baseline: We generate realistic load profiles based on simulated residents and their electric device usage. Each scenario is assigned a device profile consisting of a set of appliances, electronic devices, lighting, and other household items drawing power. Profiles are then created for individual occupant types, e.g., that cook, do other chores or are nocturnal. These occupant types can be combined to simulate the usage patterns of common household arrangements.

Each device consists of a usage profile with the device's power consumption as obtained from common device vendor websites. Each user profile then contains the times of day, number of uses, and durations of uses of each device. The time of day and duration fields each have a time granularity of one minute, giving us minute level load profiles. Previous work has shown that refrigerators loads follow roughly a 70 minute cycle and power is only drawn for half of that duration [3]. The simulated refrigerators are assigned a cycle between 60 and 70 minutes to introduce some variation into the model.

Power usage for the water heater is generated as a simplified version of the model used in [11]. Heat loss due to hot water use for showering, miscellaneous hot water usage, and ambient temperature difference is considered to decrease the water temperature at a constant rate. This results in only negligible variations in the power usage compared to the previous model. The HVAC system is simulated using a pre-calculated load curve for a given temperate pattern. The compressor is then simulated to approximately meet the load curve.

2) Legitimate Changes: Two modifications are made to the baseline load profiles: *legitimate*, and *malicious*. In the legitimate load, the traces are perturbed to reflect legitimate deviations from the baseline. Ideally, AMIDS will not raise any alerts for legitimate traces. Those traces are: (*Legit-Replace*) the replacement of a large appliance with a version 30% more efficient, (*Legit-Season*) reduced usage of heating or cooling appliances due to seasonal changes, and (*Legit-Occupant*) modified use of all appliances due to occupancy change.

3) Malicious Changes: In the malicious scenarios, the traces are perturbed to reflect load changes caused by common energy theft scenarios. Ideally, AMIDS will raise an alert for each malicious trace. The three malicious cases are: (*Mal-Bypass*) the bypassing of a large appliance, e.g., HVAC, around the meter, (*Mal-Disconnect*) periodic disconnection of the meter resulting in zero usage, and (*Mal-Reduction*) a constant reduction in measured power, e.g., due to magnets or meter hacking.

We will evaluate accuracy of the individual proposed detection solutions and the integrated AMIDS approach on various normal (baseline) and anomalous usage profiles.

B. Integrated Intrusion Detection

We implemented the proposed HMM-based solution for the integrated energy theft detection, and evaluated its overall detection capability in dealing with sensor inaccuracies. In particular, AMIDS was tested against three complete and incomplete energy theft attack attempts (see Table II). The first attack was a 5-step energy theft attack which was reported by the intrusion detection sensors accurately (each step was reported by the corresponding sensor). Each row in Table II shows the posterior distribution over the attack graph's state space. As expected, AMIDS can detect the energy theft attempt accurately, i.e., $P(s_g|observations) = 1$. During the second attack scenario (identical steps), some alerts were not triggered by the sensors, and hence AMIDS had to infer the steps based on the attack graph structure. As shown in the table, after the last step, AMIDS reports the energy theft attempt with 85% confidence. The last incomplete attack scenario which actually does not result in the goal state is not reported as a successful energy theft attempt with 87% confidence.

C. Accuracy

We now evaluate the accuracy of AMIDS under a number of attacks on a load profile for a single occupant apartment. We are particularly interested in the accuracy gains that can be

TABLE II
MULTI-SENSOR ENERGY-THEFT DETECTION USING THE AMIDS FRAMEWORK

		Attack Graph States ([Privilege|Consequence], as defined in Figure 5)																	
	Step $\mapsto$ Observation	$\varnothing\|\varnothing$	$\varnothing\|T$	$\varnothing\|TC_M$	$M\|TC_M$	$M\|TC_M I_M$	$M\|C_M I_M$	$M\|TC_M A_M$	$\varnothing\|A_N$	$\varnothing\|I_N$	$M\|\varnothing$	$M\|I_M$	$M\|A$	$\varnothing\|D$	$\varnothing\|R$	$\varnothing\|C_M$	$M\|C_M$	$M\|C_M A_M$	Goal state
Attack 1	$A_{p5} \mapsto O_7$	0	0.65	0	0	0	0	0	0.06	0.06	0.06	0	0	0.06	0.06	0.06	0	0	0
	$A_{p3} \mapsto O_{10}$	0	0	0.95	0	0	0	0	0	0	0	0	0	0	0	0	0.01	0	0.03
	$A_{c3} \mapsto O_3$	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	$A_{c2} \mapsto O_2$	0	0	0	0	0.92	0	0.08	0	0	0	0	0	0	0	0	0	0	0
	$A_{d2} \mapsto O_{14}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Attack 2	$A_{p3} \mapsto O_{10}$	0	0.14	0	0	0	0	0	0.14	0.14	0.14	0	0	0.14	0.14	0.14	0	0	0
	$A_{c2} \mapsto O_2$	0	0	0.14	0	0	0	0	0	0	0	0.07	0.07	0	0	0	0.14	0	0.57
	$A_{d2} \mapsto O_{14}$	0	0	0	0.08	0	0.04	0	0	0	0	0	0	0	0	0	0	0.04	0.85
Attack 3	$A_{p4} \mapsto O_{11}$	0	0.08	0	0	0	0	0	0.08	0.08	0.08	0	0	0.08	0.08	0.5	0	0	0
	$A_{p5} \mapsto O_7$	0	0	0.08	0	0	0	0	0	0	0	0.04	0.04	0	0	0	0.5	0	0.33
	$A_{c3} \mapsto O_3$	0	0	0	0.13	0	0.38	0	0	0	0	0	0	0	0	0	0	0.38	0.13

TABLE III
EMPIRICAL DETECTION RESULTS FOR SEVERAL ATTACK STEPS

	Cyber			Physical			Data Modification			
Detection	**Network Exploit (A_{c1})**	**Intercept Comm. (A_{c5})**	**Mal-Login (A_{c3})**	**Meter Breakin (A_{p1})**	**Meter Disconnect (A_{p3})**	**Appliance Bypass (A_{p6})**	**Legit-Replace**	**Legit-Season**	**Legit-Occupant**	**Mal-Reduction (A_{d3})**
Cyber IDSs	✓	✓	✓	-	-	-	-	-	-	-
Physical IDSs	-	-	-	✓	✓	✓	-	-	-	-
Supervised	-	-	-	-	✓	×	✓	✓	✓	✓
Unsupervised	-	-	-	-	✓	✓	✓	×	×	✓
AMIDS (HMM)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

made through information fusion of (*i*) cyber IDS alerts, (*ii*) physical tampering alerts, and (*iii*) load-based IDS alerts, as compared to the accuracy of the individual methods. Table III shows the results of running the individual IDSs as well as the combined HMM approach on a single-occupant dwelling. A check mark means that the correct action was taken, and an $\times$ indicates a false positive or false negative. A dash indicates that the experiment did not apply. As can be seen, the combined approach eliminates the false positives of the individual approaches. Alerting capabilities for the cyber and physical IDSes were validated experimentally on real meters in the TCIPG testbed [24]. In particular, we disconnected and reversed meters and checked that alerts were generated. We also collected a week of meter traffic in a mesh network of nine meters and made connection attempts towards meters using a rogue software client in order to test our implementation of the ANSI C12.22 specification-based IDS.

Of particular instances are the three **Legit** cases designed to cause false positives in the load-based approaches. Indeed, the unsupervised learning algorithm identified two as malicious behavior. The lack of any cyber or physical IDS alerts in these cases resolved these false positives in the combined approach. An additional false negative by the supervised approach was also resolved. While additional field testing is necessary, these results show that the HMM approach used by AMIDS is an effective solution for combining smart meter data sources to identify energy theft behaviors.

VII. CONCLUSIONS

In this paper, we presented AMIDS, an integrated intrusion detection solution to identify malicious energy theft attempts in advanced metering infrastructures. AMIDS makes use of different information sources to gather sufficient amount of evidence about an on-going attack before marking an activity as a malicious energy theft. Our experimental results show that through an effective information fusion and using the correlation among the triggered alerts, AMIDS can detect various types of energy theft attempts accurately using individually inaccurate sensors.

REFERENCES

[1] FBI: Smart Meter Hacks Likely to Spread. Available at http://krebsonsecurity.com/2012/04/fbi-smart-meter-hacks-likely-to-spread/.

[2] Electricity thefts surge in bad times. Available at http://www.usatoday.com/money/industries/energy/2009-03-16-electricity-thefts_N.htm.

[3] M. Armstrong, M.C. Swinton, H. Ribberink, I. Beausoleil-Morrison, and J. Millette. Sythetically derived profiles for representing occupant-driven electric loads in canadian housing. *Journal of Building Performance Simulation*, 2(1):15–30, 2009.

[4] CJ Bandim, JER Alves Jr, AV Pinto Jr, FC Souza, MRB Loureiro, CA Magalhaes, and F. Galvez-Durand. Identification of energy theft and tampered meters using a central observer meter: a mathematical approach. In *IEEE/PES Transmission and Distribution Conference and Exposition*, volume 1, pages 163–168, 2003.

[5] R. Berthier and W.H. Sanders. Specification-based intrusion detection for advanced metering infrastructures. In *IEEE Pacific Rim International Symposium on Dependable Computing*, pages 184–193, 2011.

[6] A. Cassandra. *Exact and approximate algorithms for partially observable Markov decision processes*. PhD thesis, Brown University, 1998.

[7] S. Depuru, L. Wang, and V. Devabhaktuni. A conceptual design using harmonics to reduce pilfering of electricity. In *IEEE Power and Energy Society General Meeting*, pages 1–7, 2010.

[8] S. Depuru, L. Wang, and V. Devabhaktuni. Support vector machine based data classification for detection of electricity theft. In *IEEE/PES Power Systems Conference and Exposition*, pages 1–8, 2011.

[9] S.S.S.R. Depuru, L. Wang, V. Devabhaktuni, and N. Gudi. Measures and setbacks for controlling electricity theft. In *IEEE North American Power Symposium*, pages 1–8, 2010.

[10] S.S.S.R. Depuru, L. Wang, V. Devabhaktuni, and P. Nelapati. A hybrid neural network model and encoding technique for enhanced classification of energy consumption data. In *IEEE Power and Energy Society General Meeting*, pages 1–8, 2011.

[11] C. Goh and J. Apt. Consumer strategies for controlling electric water heaters under dynamic pricing. *Carnegie Mellon Electricity Industry Center Working Paper*, 2004.

[12] G.W. Hart. Nonintrusive appliance load monitoring. *Proceedings of the IEEE*, 80(12):1870–1891, 1992.

[13] M. LeMay and C. Gunter. Cumulative attestation kernels for embedded systems. *Computer Security–ESORICS 2009*, pages 655–670, 2009.

[14] Mikhail A. Lisovich, Deirdre K. Mulligan, and Stephen B. Wicker. Inferring personal information from demand-response systems. *IEEE Security and Privacy*, 8(1):11–20, 2010.

[15] Betsy Loeff. Deputizing data: Using ami for revenue protection. *Utility Automation and Engineering*, 2008.

[16] S. McLaughlin, D. Podkuiko, and P. McDaniel. Energy theft in the advanced metering infrastructure. *Critical Information Infrastructures Security*, pages 176–187, 2010.

[17] S. McLaughlin, D. Podkuiko, S. Miadzvezhanka, A. Delozier, and P. McDaniel. Multi-vendor penetration testing in the advanced metering infrastructure. In *Proceedings of the Annual Computer Security Applications Conference*, pages 107–116. ACM, 2010.

[18] Stephen McLaughlin, Dmitry Podkuiko, and Patrick McDaniel. Energy theft in the advanced metering infrastructure. In *Proceedings of the international conference on Critical information infrastructures security*, pages 176–187. Springer-Verlag, 2010.

[19] A. Molina-Markham, P. Shenoy, K. Fu, E. Cecchet, and D. Irwin. Private memoirs of a smart meter. In *Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building*, pages 61–66, 2010.

[20] J. Nagi, KS Yap, SK Tiong, SK Ahmed, and AM Mohammad. Detection of abnormalities and electricity theft using genetic support vector machines. In *IEEE TENCON Region 10 Conference*, pages 1–6, 2008.

[21] E. Quinn. Smart metering and privacy: Existing law and competing policies. A report for the Colorado Public Utilities Commission, 2009.

[22] L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[23] Alfredo Rial and George Danezis. Privacy-Preserving Smart Metering. Technical Report MSR-TR-2010-150, Microsoft Research, 2010.

[24] T. Yardley. Keynote address: Developing a testbed for the smart grid. In *IEEE Conference on Local Computer Networks*, pages 1–1, 2010.