Preventing Occupancy Detection From Smart Meters

Dong Chen, *Student Member, IEEE,* Sandeep Kalra, *Student Member, IEEE*, David Irwin, *Member, IEEE*, Prashant Shenoy, *Fellow, IEEE*, and Jeannie Albrecht, *Member, IEEE*

***Abstract*—Utilities are rapidly deploying smart meters that measure electricity usage in real-time. Unfortunately, smart meters indirectly leak sensitive information about a home's occupancy, which is easy to detect because it highly correlates with simple statistical metrics, such as power's mean, variance, and range. To prevent occupancy detection, we propose using the thermal energy storage of electric water heaters already present in many homes. In essence, our approach, which we call combined heat and privacy (CHPr), modulates a water heater's power usage to make it look like someone is always home. We design a CHPr-enabled water heater that regulates its energy usage to thwart a variety of occupancy detection attacks without violating its objective—to provide hot water on demand—and evaluate it in simulation using real data. Our results show that a standard 50-gal CHPr-enabled water heater prevents a wide range of state-of-the-art occupancy detection attacks.**

***Index Terms*—Data privacy, Internet of things, smart grids.**

I. Introduction

UTILITIES are rapidly replacing existing electromechanical meters, which are read manually once a month, with smart meters that transmit a building's electricity usage every few minutes. In 2011, an estimated 493 utilities in the U.S. had collectively installed more than 37 million smart meters [1]. Unfortunately, smart meters also indirectly leak private, and potentially valuable, information about a building's occupants' activities [2]–[5]. To extract this information, third-party companies are now employing cloud-based, "big data" platforms to analyze smart meter data *en masse* [6]–[8].

While the purpose is, ostensibly, to provide consumers energy-efficiency recommendations, companies are mining the data for any profitable insights. For example, detecting power signatures—sequences of changes in power unique to a device—for specific appliance brands could aid manufacturers in guiding their marketing campaigns, e.g., identifying homes with General Electric versus Maytag appliances [6]. Many utilities are providing third-party companies access to troves of smart meter data. For instance, a recent report highlights one utility's practice of requiring its customers to consent to sharing their data with third parties before permitting them to use an online web portal [9]. Such privacy violations have led to a small, but growing, backlash against smart meter deployments [10].

An important example of simple and private information that smart meters leak is occupancy—whether or not someone is home and when. Tech-savvy criminals are already exploiting similar types of unintentional occupancy leaks, e.g., via publicly-visible online calendars and Facebook posts [11], to select victims for burglaries. In addition, occupancy may also indirectly reveal private information that is of interest to insurance companies, marketers, potential employers, or the government, e.g., in setting rates, directing ads, vetting an applicant's background, or monitoring its citizens, respectively. Such information could include whether a home's occupants include a stay-at-home spouse, maintain regular working hours and daily routines, frequently go on vacation, or regularly eat out for lunch or dinner.

As recent work demonstrates [12], [13], launching attacks that extract occupancy from smart meter data is surprisingly easy, since occupancy highly correlates with simple statistical metrics, such as power's mean, variance, and range. Intuitively, users' interaction with electrical devices, e.g., turning them on and off, lends itself to straightforward attacks that detect changes in these metrics and associates them with changes in occupancy. Prior work [12], [13] has observed the correlation between occupancy and power across many different homes.

Prior research proposes techniques to thwart privacy attacks on smart meter data [3], [5], [14], [15]. Broadly, these techniques use chemical energy storage, in the form of batteries, to power, or absorb, a fraction of a building's total load, thereby changing the pattern of external grid power usage the smart meter records. By carefully controlling when batteries charge and discharge, the techniques aim to prevent detecting appliance power signatures using sophisticated algorithms for nonintrusive load monitoring (NILM) [16]–[18]. However, these prior approaches do not change the statistical properties, e.g., high mean power, variance, and range, that imply occupancy, and are not designed to prevent occupancy detection. Thus, new techniques are necessary.

To address the problem, we propose combined heat and privacy (CHPr), which regulates thermal, rather than chemical,

Manuscript received May 26, 2014; revised September 22, 2014; accepted December 29, 2014. Date of publication February 26, 2015; date of current version August 19, 2015. This work was supported in part by the National Science Foundation under Grant CNS-1405826, Grant CNS-1253063, Grant CNS-1143655, and Grant CNS-0916577, and in part by the Massachusetts Department of Energy Resources. A portion of this paper appeared in a previously published conference paper [33]. Paper no. TSG-00505-2014.

D. Chen, S. Kalra, D. Irwin, and P. Shenoy are with the University of Massachusetts at Amherst, Amherst, MA 01003 USA (e-mail: irwin@ecs.umass.edu).

J. Albrecht is with the Department of Computer Science, Williams College, Williamstown, MA 01267 USA.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSG.2015.2402224

energy storage to make it look like someone is always home. In this paper, we integrate CHPr functionality into the electric water heaters already found in many homes. Water heaters effectively serve as thermal energy storage devices that CHPr can control to mask occupancy. In particular, we design a CHPr-enabled water heater with the goal of preventing occupancy detection without running out of hot water. CHPr is inspired by combined heat and power [19], which leverages the waste heat produced from generating electricity for heating buildings. Our hypothesis is that a CHPr-enabled water heater is capable of regulating its power usage to prevent occupancy detection while still providing hot water on demand. In evaluating our hypothesis, we make the following contributions.

A. Design Alternatives

We outline the design alternatives for preventing occupancy detection, including using both chemical and thermal energy storage, from smart meter data. In doing so, we review a wide range of sophisticated occupancy detection attacks based on thresholding [13], *k*-nearest neighbors (*k*-NNs), hidden Markov models (HMMs), and support vector machines (SVMs) [12].

B. CHPr-Enabled Water Heater

We present the design of our CHPr-enabled water heater and its algorithm for regulating energy usage to prevent occupancy detection without running out of hot water. Our approach combines multiple techniques to accomplish this goal.

1) It uses partial demand flattening to eliminate a large majority of power variations.
2) It injects artificial power signatures to obscure the relationship between occupancy and high, variable demand.
3) It adjusts its operation based on home activity patterns.

C. Implementation and Evaluation

We experiment with a CHPr-enabled water heater in simulation by quantifying its effectiveness using data from a prototype home and a real water heater. We show that CHPr-enabled water heaters reduce the accuracy of the occupancy detection attacks above. As one example, CHPr decreases the Matthews correlation coefficient (MCC)—a standard measure of a binary classifier's overall performance—of a threshold-based attack on the home's smart meter data by a factor of 10 (from 0.44 to 0.045). In addition, we also show that, even though CHPr was not designed to prevent NILM [16], [18], it actually outperforms prior battery-based techniques at reducing the accuracy of a state-of-the-art NILM algorithm without requiring the use of expensive batteries.

II. Background

We assume a building equipped with a smart meter that records average power $P(t)$ over a sampling interval T, yielding a time-series of power values. Today's newer utility-grade smart meters support sampling intervals from 1 to 5 min, while older meters support 15 min to 1 h. Thus, we focus on preventing occupancy detection from smart meters with a one-minute sampling interval. Adapting our techniques to higher resolution power meters, e.g., 1 Hz or greater, is future work. We represent occupancy as a binary function $O(t)$, over each sampling period t, where zero represents an unoccupied home and one represents a home with at least one person in it. This paper focuses on masking occupancy to prevent inferring $O(t)$ from $P(t)$.

Since there is no general metric that applies to any possible occupancy detection attack, we evaluate CHPr using a threat model based on a wide range of sophisticated occupancy detection attacks. These attacks are the focus of [12] and [13] and have been shown to accurately detect occupancy across a variety of homes. With the exception of the thresholding attack, the attacks below require ground truth data to train a classifier that learns an association between occupancy and power. For the latter three attacks, we implement the attack based on details from [12].

A. Thresholding

The thresholding attack signals occupancy if power's mean, variance, or range exceeds some predefined threshold [13]. In particular, we define an epoch length T_{epoch}, and then compute power's mean, variance, and range over each epoch. In our experiments, we use 15 min as the epoch length. Anytime one of the metrics exceeds a predefined threshold, we record a potential occupancy point, resulting in a series of points in time. We then cluster points to infer occupancy over time, such that if two points are within a time threshold, e.g., 1 h, we consider the home occupied during the interval between those points.

B. k-NNs

The *k*-NN attack uses a simple *k*-NN classifier. As above, the metrics are power's mean, variance, and range every 15 min. *k*-NN effectively plots the training data in a 3-D feature space with each point labeled as either occupied or unoccupied. New data points are then classified based on which label is most frequent among the k nearest points using the Euclidean distance function. For our experiments, we set k equal to one, such that we classify new points-based solely on the label of the nearest data point. As in prior work, we implement the classifier in MATLAB [12].

C. SVMs

As with *k*-NN, SVMs plot the metrics in a 3-D feature space with each point labeled as occupied or unoccupied. However, linear SVMs compute a hyperplane that best separates data points into their respective classes, e.g., by maximizing the distance between they hyperplane and the nearest data point in any class. The separation effectively assigns each region of the 3-D space as either being occupied or unoccupied. The SVM then simply assigns new data points based on which region of the space the point resides in. To train our SVM we use libSVM [20] with a radial basis function kernel and default parameters.

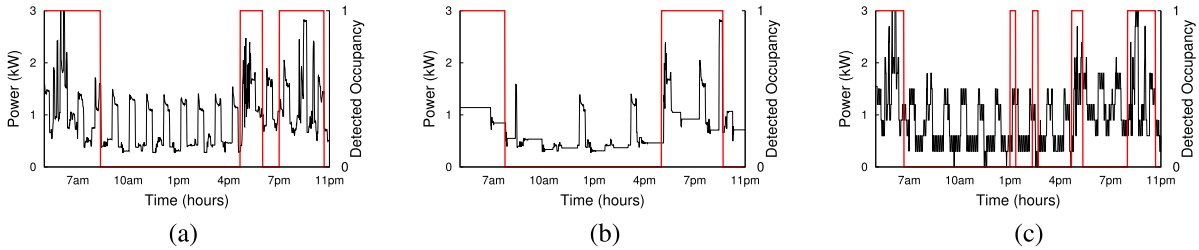

Fig. 1. (a) Threshold-based attack is effective at detecting occupancy in smart meter data when altered by BLH techniques, such as (b) NILL, or (c) LS2.

D. HMMs

Finally, we use a simple HMM that associates hidden, e.g., unknown, states with occupancy (0 for unoccupied and 1 for occupied) and visible states with discretized levels of power consumption. We characterize the HMM with two sets of probabilities learned during training: transition and emission probabilities. Transition probabilities characterize the probability of transitioning from one value of the hidden state to another, while emission probabilities indicate the probability of emitting a particular power level given a particular occupancy state (0 or 1). During classification, the transition and emission probabilities are used to assign values to the hidden states based on power readings. We implement our HMM classifier using MATLABs built-in HMM functions. In contrast to the methods above, HMM only uses average power every 15 min (and not variance and range) for training and classification. As in [12], since our power readings are continuous and HMMs require discretized power levels, we discretize power by log-binning the training and test data into 20 bins.

E. Prior Work

Prior techniques propose to alter grid power usage by controlling battery charging and discharging, called battery-based load hiding (BLH) [3], [5], [14], [15], to obscure smart meter data. BLH techniques focus on preventing NILM [16], [18], which analyzes changes in $P(t)$ to compute a separate power time-series $p_i(t)$ for each $i = 1 \ldots n$ appliances in a home. While no BLH techniques have been explicitly designed to prevent occupancy detection, we use existing BLH techniques as "strawmen" for comparison, since NILM algorithms implicitly provide occupancy information by revealing the usage of interactive appliances, such as a microwave or television.

Thus, any technique designed to prevent NILM might also prevent occupancy detection by preventing the detection of interactive appliance activity. Since there are no prior techniques to thwart occupancy detection, we choose techniques that prevent NILM as our baseline for comparison. We describe two representative examples of BLH below. As we show in Section V, while CHPr does not explicitly focus on preventing NILM, it effectively does so as a side-effect of preventing occupancy detection, outperforming the BLH techniques below without requiring the use of expensive batteries.

1) Nonintrusive Load Leveling (NILL): NILL [3] removes changes in $P(t)$ that reveal appliance power signatures by leveling, or flattening, the home's external grid demand recorded by the smart meter. In essence, NILL charges batteries when actual demand is below a target demand, and then discharges batteries when it is above the target demand, to maintain meter readings as near to the target as possible. Ideally, demand is flat and always equal to the target demand, thereby revealing only the home's average power usage and nothing else. Note that there is nothing in the design of NILL that is specific to NILM (or any particular NILM algorithm): only revealing a building's average power would also effectively prevent occupancy detection or any other information leakage. Unfortunately, for practical battery capacities, NILL diverges from this ideal. As we show, once NILL discharges its battery, it can no longer alter grid demand. Since battery depletion often occurs during the high demand periods that correlate with occupancy, NILL does not prevent occupancy detection.

2) Lazy Stepping (LS): LS [5] is an improvement to NILL that requires less battery capacity to obscure appliance power signatures from NILM. The idea behind LS is that, rather than flatten grid demand, it controls battery charging and discharging to transform demand into a step function that removes the fine-grained changes in power claimed to be useful in identifying appliances. However, as we show, LS does not prevent occupancy detection: the periods of high demand that strongly correlate with occupancy remain identifiable.

Fig. 1 visually demonstrates the points above by showing the performance of the thresholding occupancy detection attack, even after demand has been altered by NILL and LS2.[1] The graphs overlay a home's average power usage every minute with the results of our occupancy detection attack for a representative day in a real home. Fig. 1(a) shows that, for the unaltered demand, with the exception of two brief periods, the attack's predicted occupancy nearly exactly matches the ground truth, where occupants are away from 8 A.M. to 4 P.M.

Fig. 1(b) then shows the results of the same attack on demand altered by NILL using a 6 kWh battery, as in [3]. Despite the altered demand, the attack is still able to accurately detect occupancy. The NILL-altered demand demonstrates that, in practice, battery capacity limitations prevent ideal demand flattening that obscures occupancy detection. As expected, NILL does not prevent the high demand periods that correlate with occupancy, since it tends to deplete its battery

[1]LS2 is the best performing variant of LS [5].

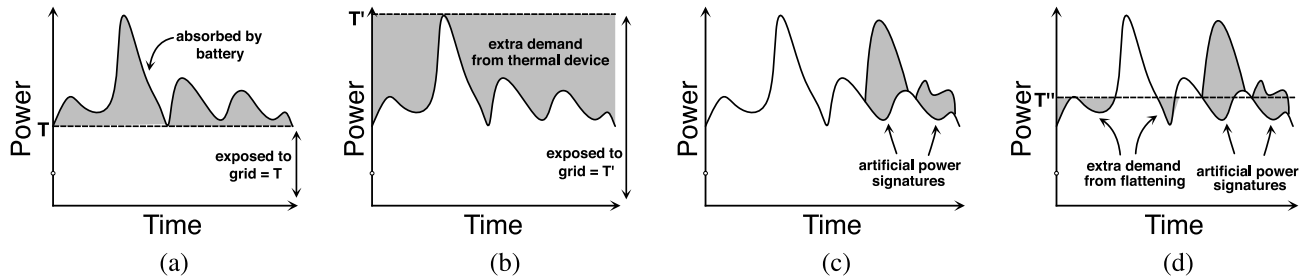

Fig. 2. Different options for masking occupancy, including (a) demand flattening using both BLH, (b) thermal energy storage, (c) artificial power signature injection, and (d) CHPrs hybrid approach, that combines demand flattening and artificial signature injection to minimize its energy requirements.

TABLE I
VALUES FOR THE MCCS OVER THE SAME WEEK AS IN FIG. 5

	Threshold	*KNN*	*HMM*	*SVM*
Original (a)	0.44	0.17	0.51	0.37
NILL (b)	0.41	0.19	0.46	0.36
LS2 (c)	0.43	0.012	0.56	0.40

during these periods, eliminating the option to later discharge its batteries to mask high demand. Of course, there exists a larger battery capacity, such that NILL would completely flatten demand at the average, thereby preventing occupancy detection. However, 6 kWh of capacity[2] already imposes an excessively high cost—$708 per year amortized over a battery's lifetime based on recent cost estimates [21], which would increase an average U.S. home's annual electricity bill by roughly 50% [22].

Likewise, Fig. 1(c) shows the results of the attack on demand altered by the LS2 algorithm, which uses much less battery capacity—0.5 kWh in this case, as in [5]—than NILL. As the graph demonstrates, with 0.5 kWh of battery capacity, LS2s battery is simply too small to mask the periods of high demand by discharging its battery. Instead, LS2 discretizes demand to obscure the many small changes in power that NILM might leverage to identify appliances. As Fig. 1(c) shows, due to the small capacity battery, demand altered by LS2 retains the general shape of the original demand profile including the periods of high, variable demand that indirectly reveal the home's occupancy status.

Table I quantifies the effectiveness of all of our attacks over the same week as in Fig. 5 by showing the MCC [23], a standard measure of a binary classifier's performance, where values are in the range -1.0 to 1.0, with 1.0 being perfect detection, 0.0 being random prediction, and -1.0 indicating detection is always wrong. MCC values closer to 0.0, or random prediction, are better for masking occupancy. The expression for computing MCC is below, where TP is the fraction of true positives, FP is the fraction of false positives, TN is the fraction of true negatives, and FN is the fraction of false negatives, such that TP + FP + TN + FN = 1. The table shows that neither NILL nor LS2 significantly lowers the MCC of the thresholding, HMM, and SVM occupancy detection attacks. While LS2 reduces the detection accuracy of the *k*-NN attack, the results show that *k*-NN is the worst performing and most unsophisticated attack

$$\frac{\mathrm{TP} * \mathrm{TN} - \mathrm{FP} * \mathrm{FN}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})}}. \quad (1)$$

3) Summary: Our results show that existing BLH techniques do not prevent occupancy as a side-effect of attempting to prevent NILM. In addition, any BLH technique wastes a significant fraction of any energy it stores in its battery, due to energy conversion losses. These losses are at least 20% of the stored energy with existing battery and inverter technology [24]. The insights above lead to CHPrs approach, which leverages the thermal energy storage in large elastic heating loads, such as water heaters, to cheaply and efficiently mask occupancy. In addition, since CHPr only reschedules energy a water heater already consumes, it avoids conversion losses.

III. USING THERMAL STORAGE: DESIGN ALTERNATIVES

We consider the design alternatives for using thermal energy storage to mask occupancy. Fig. 2 highlights the differences between BLH and thermal energy storage. BLH flattens grid demand by controlling battery charging and discharging, such that, in the ideal (although not in practice for reasonable battery capacities), the smart meter always sees a steady, flat power consumption level [depicted by T in Fig. 2(a)]. Whenever the home's demand rises above T, BLH discharges its battery to provide the home additional power, rather than drawing it from the grid. The approach thwarts occupancy detection attacks by "clipping" any power usage above T, exposing a constant power usage to the smart meter that effectively makes it look like no one is ever home.[3]

Thermal energy storage is also capable of flattening demand in a similar manner, although it cannot "clip" power usage in the same way as a battery, since it is incapable of discharging general-purpose electricity, i.e., it cannot convert its heat back into electricity. Instead, thermal energy storage can only flatten demand by raising grid power usage, e.g., by converting electricity into heat, to its peak level [depicted by T' in Fig. 2(b)]. In this case, the thermal storage device controls its resistive heating elements to draw a variable amount of power (above the normal power draw) to ensure that the total power draw is always T'. Thus, thermal energy storage is able to thwart occupancy detection by "boosting" power usage such

[2]Cost estimates are based on a commercially-available sealed absorbed glass mat/valve-regulated lead-acid deep-cycle lead–acid battery designed for home solar panel installations.

[3]An occupancy detector may still detect occupancy, if T is sufficiently high.

that the home always draws a steady power T' from the grid. The thermal device then stores the heat for later use.

Since the homes we monitor have a high peak-to-average power ratio, raising power usage to the peak value T' requires a substantial amount of energy, which in turn requires a large amount of thermal energy storage capacity to make use of the heat. To reduce the power necessary to mask occupancy, thermal energy storage can also leverage artificial power signature injection, which controls the thermal device to inject "noise" that resembles real electrical loads in the home [depicted in Fig. 2(c)]. By injecting fake signatures that resemble real loads during low-power periods when no one is home, the approach makes it appear that someone is always home, which also thwarts occupancy detection, but using less energy. As before, the thermal device stores its heat for later use. As we describe in the next section, CHPr leverages a hybrid approach [in Fig. 2(d)] that combines artificial signature injection with partial demand flattening, such that it raises demand to an intermediate value T'' (below the peak value T'). Since partial demand flattening reveals peaks above T'', CHPr only injects signatures larger than T''.

IV. CHPr-Enabled Water Heater

A standard tank-based residential water heater includes a reserve tank with a cold-water inlet pipe at the bottom and a hot-water outlet pipe at the top, since heated water naturally rises to the top of the tank. Residential water heaters include tanks that range in size from 30–100 gal (equivalent to 113.6–378.5 L, respectively) with heating elements ranging from 3500 to 5500 W. Importantly, a water heater's average total energy usage (and its thermal energy capacity) is a significant fraction of an average home's usage. For example, a standard 50 gal (or 189.3 L), 4.5 kW water heater that runs for three hours each day consumes 13.5 kWh [25], while an average U.S. home consumes only ∼24 kWh per day [22].

A typical water heater operates by always attempting to ensure that: 1) the tank is full and 2) the tank's water temperature is equal to an adjustable target temperature that is typically set between 120 and 140 °F (or 48.9 to 60 °C). Thus, when hot water is drawn from the tank, e.g., due to someone taking a hot shower, the water heater refills the tank with cold water, and then immediately begins heating it at maximum power until the tank's water reaches the target temperature. The temperature of the intake water is usually in the range of 50–60 °F (or 10–15.6 °C), but is dependent on the climate. Water heaters generally employ a tight guardband of 15 °F (or 8.33 °C), such that if no hot water is drawn out, the water heater waits until the water is, for example, 105 (or 40.6 °C) before reheating it to the 120 °F (or 48.9 °C) target [26]. Since hot water rises, water heaters often employ two heating elements and thermostats, one at the top and bottom of the tank.

A CHPr-enabled water heater works by relaxing the operational requirements above and not always using the maximum power to immediately heat intake water. As an example, Fig. 3 shows the power usage of a 50 gal (or 189.3 L), 4500 W water heater over one day on the left y-axis. The short regular bursts of power are due to maintaining the water temperature within

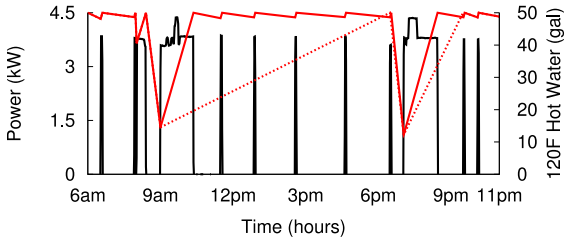

Fig. 3. Day's power usage (black) for a 50 gal (or 189.33 L), 4.5 kW water heater, and the remaining hot (120 °F/48.9 °C) water in its tank (red).

the 15 °F guardband, while the longer periods of power usage stem from heating the cold intake water that is replacing hot water drawn out of the tank. The right y-axis shows the amount of available hot water (at 120 °F), assuming ideal insulation where it takes 2.93×10^{-4} kWh to raise 1 lb (or 0.45 kg) of water by 1 °F (or 0.56 °C). We then compute the amount of 120 °F (or 48.9 °C) hot water by correlating the heater's energy usage with a volume of heated water. Fig. 3 indicates that, on this day, the tank never runs out of hot water. The figure also shows that the water heater could heat at a slower constant rate (indicated by the dotted red lines) using less than the maximum power without ever running out of hot water. Rather than heat at a slow constant rate, CHPr varies the heating element's power usage to partially flatten demand and inject artificial signatures, while using the same amount of energy over the period.

To determine how fast it must heat water to prevent running out, which dictates the energy it must consume over a given period, CHPr tracks the amount of remaining hot (120 °F/48.9 °C) water at the top of the tank and estimates the time until the next significant use of hot water. Our current implementation simply maintains an estimate of the average length t between usage periods greater than 25 gal (or 94.66 L), or roughly a single shower, and ensures that after a significant usage period all the water is heated within t. While more sophisticated methods for estimating t are possible, we did not explore them since our simple method proved effective. Given an energy budget and this time period estimate t, CHPr then determines how much to partially flatten demand and inject artificial signatures, as described below.

A. Partial Demand Flattening

Since a water heater does not use enough energy to completely flatten demand at its peak, CHPr employs a flattening threshold P_{flat} that only partially flattens demand to a target level less than the peak demand. To maintain P_{flat} at each t with current demand $N(t)$, CHPr consumes P_{flat}-$N(t)$ whenever $N(t) < P_{\text{flat}}$. Since average demand is typically much lower than peak demand, a low flattening threshold is able to hide a large percentage of the changes in power without using much energy.

B. Artificial Power Signature Injection

Partially flattening demand still exposes changes in power that occur above the threshold. To hide these changes, CHPr injects artificial power signatures. Importantly, CHPr does not simply inject demand randomly, since an attacker may be able to detect these random or atypical patterns in smart meter data.

Instead, CHPr replays realistic power signatures. These power signatures are derived from the home's aggregate data, by storing, in a database, sequences of the home's power changes that occur above the flattening threshold. CHPr also takes additional steps to ensure artificial demand is difficult to discern from real demand. For example, the power signature database includes attributes for each signature, such as average power and duration. CHPr then divides power signatures into categories based on their attributes, e.g., small, medium, large and short, medium, and long, and computes the fraction of signatures in each category.

We use this fraction to weight each category's random selection, such that the artificial demand matches the breakdown of real demand. In addition, to prevent attackers from detecting repeated signatures, CHPr introduces some randomness into the replayed signature by raising or lowering each point by a small random amount, e.g., 0%–5% of usage. To further reduce its energy requirements, CHPr only injects signatures when the home is unoccupied. Our premise is that injecting artificial power signatures should not be necessary when a home is occupied—there is no need to make the data look like someone is home when someone actually is home. When the home is unoccupied, CHPr randomly selects signatures from the database to inject and replay at an injection rate equal to the rate at which the home generates power signatures above the flattening threshold when occupied. Our prototype explicitly tracks home occupancy by monitoring occupants' GPS coordinates in real time via a smartphone application.

Finally, CHPr also adjusts its flattening threshold and rate of artificial signature injection over time to match the expected rate each period. Our premise is that there is no need to make low-power nighttime periods look like high-power daytime periods, or low-power weekdays look like high-power weekends. Instead, CHPr need only ensure these time periods look the same with respect to each other, regardless of whether a home is occupied or unoccupied. Thus, CHPr indexes its power signature database based on each signature's real time-of-use. At any time, CHPr randomly selects from past power signatures that occurred near that time, e.g., within an hour, since typical power signatures in the morning, e.g., a coffee maker, are likely to be different from those in the evening, e.g., a TV. Indexing signatures by time is also important because an attacker could exploit usage patterns that appear unnatural.

C. Tuning CHPr

CHPr sets the flattening threshold P_{flat} for each period based on the excess energy available after estimating the energy required to inject artificial signatures (based on the rate of signatures observed when the home is occupied). Of course, CHPr could run out of energy if its estimated energy budget over a time period t is inaccurate or occupants leave for extended periods, such that the water heaters does not have enough thermal capacity to partially flatten demand and inject artificial signatures over the period. As with BLH, whenever CHPr runs out of energy it has no choice but to expose the home's raw usage to the smart meter. We evaluate the frequency and impact of running out of energy in Section V.

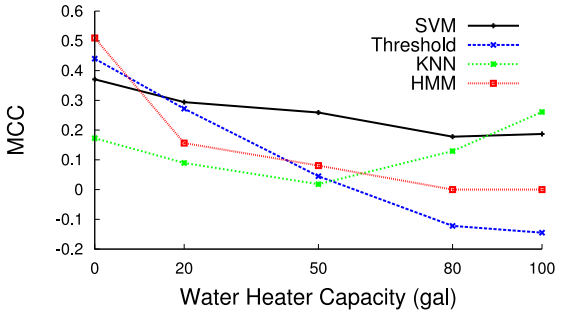

Fig. 4. CHPr decreases the MCC of the occupancy detection attacks as water heater size increases, with the exception of the k-NN attack with a water heater size greater than 50 gal.

V. Evaluation

We implement a simulator in *R* to evaluate CHPr. The simulator takes as input a home's aggregate power trace and its water heater power trace, and reschedules the water heater's power consumption based on the approach outlined in the previous section. The simulator uses minute-level power data we have gathered from a real home; our home deployment and our data is described in detail in [27]. We also require ground truth occupancy to quantify the performance of our occupancy detection attacks.

To gather ground truth occupancy data, the adult occupants in the home run a real-time geolocation application on their cell phone, which we query to determine the home's ground truth occupancy (based on the occupants' GPS coordinates). We have collected GPS data for roughly one year, and power data for three months; we train our classifiers below (and from Section II) on 82 days worth of data. Note that, while our home's occupancy rate may appear high, based on our own data collection at other homes and national statistics [22], [28], we believe the power usage and occupancy pattern are representative of a large class of homes. For instance, consider that even if all occupants are away for a standard 40-h work week (8 h per day), and home otherwise, the resulting occupancy rate is still 76.2% (128 out of 168 h).

A. Preventing Occupancy Detection

We evaluate CHPrs effectiveness against each of the occupancy detection attacks from Section II. We quantify the performance of the occupancy detection attack on both the original demand and the CHPr-modified demand in terms of the MCC [23]. Recall from Section II, that the MCC is a standard measure of a binary classifier's performance, where values are in the range -1.0 to 1.0, with 1.0 being perfect detection, 0.0 being random prediction, and -1.0 indicating detection is always wrong. MCC values closer to 0.0, or random prediction, are better for masking occupancy.

Fig. 4 shows how the MCC for each occupancy detection attack varies based on the capacity of the water heater. The water heater's capacity determines the amount of thermal energy storage available for CHPr. In this case, a value of 0 for the capacity indicates the performance of the attack on the original demand. The experiment shows that, with the exception of the k-NN attack, the accuracy of each occupancy

TABLE II
PERFORMANCE OF THE BEST OCCUPANCY DETECTION ATTACK (HMM) ON A WEEK OF REPRESENTATIVE DATA FROM A HOME, COMPARED WITH THE PERFORMANCE OF EACH ATTACK ON DATA MODIFIED BY CHPR WITH A 50-GAL WATER HEATER

Type	*True Positives*	*True Negatives*	False Positives	*False Negatives*	MCC
Original (HMM)	61.83%	17.86%	**6.25%**	14.06%	**0.510**
CHPr (Threshold)	73.01%	2.78%	**13.22%**	10.99%	**0.045**
CHPr (HMM)	5.58%	23.44%	**0.67%**	70.31%	**0.081**
CHPr (KNN)	49.33%	8.93%	**15.18%**	26.56%	**0.018**
CHPr (SVM)	50.67%	15.18%	**8.93%**	25.22%	**0.259**

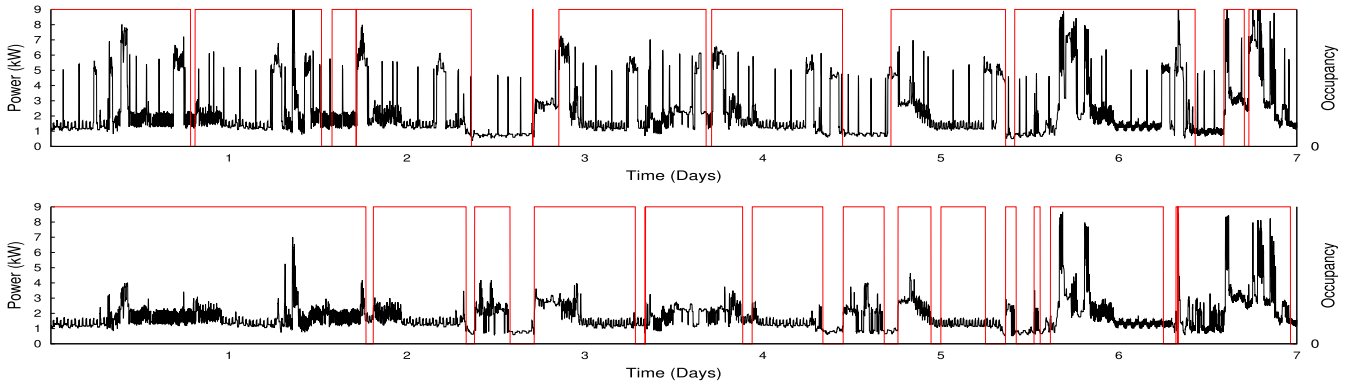

Fig. 5. Top: home's original week-long power usage and ground truth occupancy. Bottom: power usage when using a CHPr-enabled water heater and detected occupancy when using the threshold-based occupancy detection attack outlined in Section II.

detection attack decreases as the size of the water heater increases. This is somewhat expected, since the larger the water heater size, the more energy CHPr has to shift in time to mask occupancy. However, the *k*-NN attack's performance actually increases for the 80 and 100-gal sizes due to our activity optimization. The reason is that, at these sizes, concentrating signature injections in the unoccupied periods actually increases the power level during these periods beyond the occupied period. As a result, the *k*-NN classifier begins to associate these high levels of power usage with unoccupied periods. While we designed CHPr for standard-sized water heaters, where energy capacity is limited, this result suggests modifications to the activity optimization are necessary for high capacity water heaters, such that they inject energy during both occupied and unoccupied periods.

Another interesting insight from our results is that, while the threshold, SVM, and HMM attack have similar MCCs on the original power data, their performance diverges under CHPr as the water heater size increases. SVM is the most robust to CHPr, decreasing from an MCC near 0.38 to 0.259 for a 50-gal tank, while thresholding is the least robust, decreasing from an MCC of 0.45 to 0.04 for a 50-gal tank. The HMM attack has performance in between SVM and thresholding. Table II shows the breakdown of true positives, true negatives, false positives, and false negatives, as well as the MCC, for the original demand (when using HMM for detection) and each of the attacks when using a standard-sized 50 gal water heater. In this case, HMM performs the best on the original demand. The table shows that CHPr effectively reduces the MCC to near 0 for the Threshold and HMM attack.

Fig. 5 provides a visual depiction of CHPrs ability to mask occupancy using data from our home over a representative week in the summer. The top graph shows both the home's power usage each minute, including a standard 50 gal (or 189.3 L) water heater, as well as its ground truth occupancy using the occupants' GPS coordinates. The brief spikes in electricity usage throughout the week are due to heating water. The lower graph then shows the power usage after rescheduling the water heater's power consumption using CHPr, as well as the detected occupancy of this modified power trace using our threshold-based attack. A good example of CHPrs capabilities occurs between days four and five when the home is unoccupied for an extended period. Using the original demand, the low power usage clearly indicates the occupants are away, while the CHPr-modified demand makes the power usage appear similar to an occupied home. While there are a few instances where the water heater runs out of energy, i.e., fully heats all of its tank's water, that cause it to expose a low power usage that may reveal an unoccupied home, e.g., between days two and three, the data exposes much less occupancy information overall. In addition, there are no instances where our (simulated) reserve tank runs out of hot water due to CHPrs operation.

B. Preventing NILM

As we discuss in Section II, BLH techniques do not prevent occupancy detection as a side-effect of attempting to prevent NILM. Here, we show that CHPr actually outperforms NILL and LS2 in preventing NILM simply as a side-effect of preventing occupancy detection. For this experiment, we evaluate NILL, LS2, and CHPr using a state-of-the-art NILM algorithm based on factorial HMMs [29]–[31]. The algorithm is the basis for the reference energy disaggregation dataset [30] and is implemented as part of the open-source NILM Toolkit (NILM-TK) [31]. A recent paper shows that the

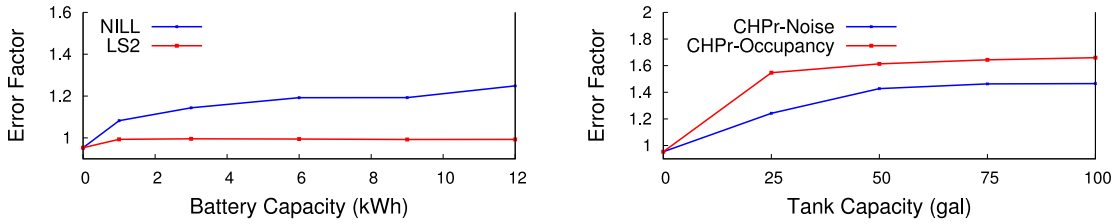

Fig. 6. Left: NILM error factor as a function of energy capacity for different methods of altering grid demand. Right: when using CHPr-based techniques.

algorithm performs outperforms other, previously proposed, NILM algorithms [31].

Note that prior work on NILL and LS2 did not evaluate their performance against a real NILM algorithm. Instead, they used general privacy metrics, such as mutual information measures, entropy, and "sister pairs," to indirectly evaluate performance. Our results indicate that these indirect measures do not necessarily correlate with the performance of a real NILM algorithm. This is due to the fact that NILM accuracy is largely dictated by the accuracy of detecting large loads, which NILL and LS2 do not effectively prevent. NILM algorithms continue to perform poorly at detecting small loads, even without actively obscuring the demand.

We quantify NILM accuracy by computing the sum of the errors between each load's actual and inferred power usage, normalized by a home's total energy usage $P(t)$, at each time t. Formally, if $\tilde{p}_i(t)$ denotes load p_is actual power usage at time t and $p_i(t)$ denotes its inferred power usage from NILM at time t, then we define an error factor δ over T intervals as

$$\delta = \frac{\sum_{i=1}^{N}\sum_{t=1}^{T}|\tilde{p}_i(t) - p_i(t)|}{\sum_{i=1}^{N}\sum_{t=1}^{T}\tilde{p}_i(t)}. \quad (2)$$

Here, the numerator is the sum of the absolute errors at each data point, and the denominator is the home's total energy usage over T. Lower values of δ are better; an error factor of zero indicates perfect NILM. While there is no upper bound on the error factor, a value of one indicates the reading-to-reading errors are equal to the home's energy usage. Note that this metric is a variant of the "total energy correctly assigned" metric from [30].

Fig. 6(left) shows the error factor for our NILM algorithm when NILL and LS2 alter grid power as a function of available battery capacity. The graph shows that when not using NILL or LS2 the error factor is 0.95 (the data point at $x = 0$). As expected, the error factor rises when using NILL, since more battery capacity enables it to flatten demand and remove more variations in power that reveal appliance usage. The rise in error factor demonstrates that NILL is reducing the accuracy of our NILM algorithm. However, notice that the rise is only 25% (from 0.95 to 1.24) when using a 12 kWh battery, which would have an amortized cost of $1416 per year based on the estimates from Section II. By contrast, LS2 is nearly completely ineffective at reducing the accuracy of our NILM algorithm; the algorithm's accuracy at recognizing appliance power signatures remains nearly the same.

Fig. 6(right) shows results for the same time period when using two versions of CHPr as a function of the water heater's reserve tank capacity. CHPr-noise uses the available energy from the water heater to inject random power values similar to [32], while CHPr-occupancy is our CHPr algorithm for preventing occupancy detection. Both CHPr-noise and CHPr-occupancy increase the error in the NILM algorithm more than NILL and LS2. Even though it is not designed to prevent NILM, CHPr-Occupancy actually performs the best, with an error factor near 1.55 for a 25-gal water tank, which has a thermal energy capacity of 6.75 kWh. With a similar size battery, NILL only achieves an error factor of 1.19. Of course, many homes already have water heaters, which could easily integrate CHPr functions, while few have any integrated battery storage, which would require a large capital investment to install.

VI. Conclusion

This paper presents CHPr, which prevents occupancy detection using the thermal energy storage inherent to the large elastic heating loads already present in many homes, in particular electric water heaters. As we show in Section II, CHPr leverages thermal energy storage to mask occupancy because using chemical energy storage, in the form of batteries, requires a level of energy storage capacity that is prohibitively expensive. CHPrs algorithm combines partial demand flattening, artificial power signature injection, and activity- and occupancy-aware optimizations to reduce its energy requirements. Importantly, CHPr does not waste any energy or increase electricity costs: it simply reschedules the energy a water heater already consumes to mask occupancy, while ensuring the reserve tank does not run out of hot water. We evaluate CHPr against multiple sophisticated occupancy detection attacks-based k-NN clustering, HMMs, SVMs, and thresholding. Our evaluation shows that CHPr is effective at masking occupancy by regulating the power usage of a standard 50 gal (or 189.3 L) water heater, decreasing the MCC of occupancy detection from 0.51 (with the HMM attack) to 0.045 with the threshold-based attack, 0.081 with the HMM attack, and 0.259 with the SVM attack.

References

[1] (2011). *U.S. Energy Information Administration, Frequently Asked Questions, How Many Smart Meters are Installed in the U.S. and Who has Them?* [Online]. Available: http://www.eia.gov/tools/faqs/faq.cfm?id=108&t=3

[2] P. McDaniel and S. McLaughlin, "Security and privacy challenges in the smart grid," *IEEE Security Privacy*, vol. 7, no. 3, pp. 75–77, May/Jun. 2009.

[3] S. McLaughlin, P. McDaniel, and W. Aiello, "Protecting consumer privacy from electric load monitoring," in *Proc. 18th ACM Conf. Comput. Commun. Security (CCS)*, Chicago, IL, USA, 2011, pp. 87–98.

[4] A. Molina-Markham, P. Shenoy, K. Fu, E. Cecchet, and D. Irwin, "Private memoirs of a smart meter," in *Proc. 2nd ACM Workshop Embedded Sens. Syst. Energy-Efficien. Build. (BuildSys)*, Zurich, Switzerland, 2010, pp. 61–66.

[5] W. Yang *et al.*, "Minimizing private data disclosures in the smart grid," in *Proc. ACM Conf. Comput. Commun. Security (CCS)*, Raleigh, NC, USA, 2012, pp. 415–427.

[6] (May 2013). *Bidgely*. [Online]. Available: http://bidgely.com

[7] (May 2013). *Chai Energy*. [Online]. Available: http://www.mychai.co/

[8] (May 2013). *PlottWatt*. [Online]. Available: https://plotwatt.com/

[9] V. Chadwick, C. Butt, and H. Cook, "Smart meter data shared far and wide," The Age, Docklands, VIC, Australia, Sep. 2012. [Online]. Available: http://www.theage.com.au/it-pro/government-it/smart-meter-data-shared-far-and-wide-20120922-26dvp.html

[10] (May 2013). *Stop Smart Meters!* [Online]. Available: http://stopsmartmeters.org/

[11] A. Bloxham. (Sep. 2011). *The Telegraph, Most Burglars Using Facebook and Twitter to Target Victims, Survey Suggests*. [Online]. Available: http://www.telegraph.co.uk/technology/news/8789538/Most-burglars-using-Facebook-and-Twitter-to-target-victims-survey-suggests.html

[12] W. Kleiminger, C. Beckel, T. Staake, and S. Santini, "Occupancy detection from electricity consumption data," in *Proc. 5th ACM Workshop Embedded Syst. Energy-Efficient Build. (BuildSys)*, Rome, Italy, 2013, pp. 1–8.

[13] D. Chen, S. Barker, A. Subbaswamy, D. Irwin, and P. Shenoy, "Non-intrusive occupancy monitoring using smart meters," in *Proc. 5th ACM Workshop Embedded Syst. Energy-Efficient Build. (BuildSys)*, Rome, Italy, 2013, pp. 1–8.

[14] M. Backes and S. Melser, "Differentially private smart metering with battery recharging," *IACR Cryptol.*, no. 183, pp. 1–28, Apr. 2012. [Online]. Available: https://eprint.iacr.org/2012/183

[15] G. Kalogridis, C. Efthymiou, S. Denic, T. Lewis, and R. Cepeda, "Privacy for smart meters: Towards undetectable appliance load signatures," in *Proc. Int. Conf. Smart Grid Commun. (SmartGridComm)*, Gaithersburg, MD, USA, 2010, pp. 232–237.

[16] K. Armel, A. Gupta, G. Shrimali, and A. Albert, "Is disaggregation the holy grail of energy efficiency? The case of electricity," *Energy Policy*, vol. 52, no. 1, pp. 213–234, Jan. 2013.

[17] G. Hart, "Nonintrusive appliance load monitoring," *Proc. IEEE*, vol. 80, no. 12, pp. 1870–1891, Dec. 1992.

[18] M. Zeifman and K. Roth, "Nonintrusive appliance load monitoring: Review and outlook," *IEEE Trans. Consum. Electron.*, vol. 57, no. 1, pp. 76–84, Feb. 2011.

[19] *Combined Heat and Power: A Clean Energy Solution*, Environ. Protect. Agency, San Francisco, CA, USA, Aug. 2012.

[20] C. Chang and C. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 27, Apr. 2011, Art. ID 27.

[21] A. Mishra, D. Irwin, P. Shenoy, J. Kurose, and T. Zhu, "SmartCharge: Cutting the electricity bill in smart homes with energy storage," in *Proc. Future Energy Syst. Energy Comput. Commun. Meeting (e-Energy)*, Madrid, Spain, 2012, pp. 1–10.

[22] D. Cauchon. (Dec. 2011). *USAToday, Household Electricity Bills Skyrocket*. [Online]. Available: http://www.usatoday.com/money/industries/energy/story/2011-12-13/electric-bills/51840042/1

[23] B. Matthews, "Comparison of the predicted and observed secondary structure of T4 phage lysozyme," *Biochim. Biophys. Acta.*, vol. 405, no. 2, pp. 442–451, Oct. 1975.

[24] S. Schoenung, "Energy storage systems cost update: A study for the DOE energy storage systems program," Dept. Energy, Sandia Nat. Lab., Albuquerque, NM, USA Tech. Rep. SAND2011-2730, Apr. 2011.

[25] (May 2013). *WaterHeaterTimer.org*. [Online]. Available: http://waterheatertimer.org/How-much-does-it-cost-to-run-water-heater.html

[26] (Dec. 2014). *Water Heaters are not Water Delivery Temperature Control Devices*. [Online]. Available: http://www.cashacme.com/legionella_related_info_art1.php

[27] S. Barker *et al.*, "Smart*: An open data set and tools for enabling research in sustainable homes," in *Proc. Workshop Data Min. Appl. Sustain. (SustKDD)*, Beijing, China, 2012, pp. 1–6.

[28] *Electric Power Monthly With Data for February 2013*, U.S. Energy Inf. Admin., Washington, DC, USA, Apr. 2013.

[29] H. Kim, M. Marwah, M. Arlitt, G. Lyon, and J. Han, "Unsupervised disaggregation of low frequency power measurements," in *Proc. SIAM Int. Conf. Data Min. (SDM)*, Columbus, OH, USA, 2011, pp. 747–758.

[30] J. Kolter and M. Johnson, "REDD: A public data set for energy disaggregation research," in *Proc. Workshop Data Min. Appl. Sustain. (SustKDD)*, San Diego, CA, USA, 2011, pp. 1–6.

[31] N. Batra *et al.*, "NILMTK: An open source toolkit for non-intrusive load monitoring," in *Proc. 5th Int. Conf. Future Energy Syst. (e-Energy)*, Cambridge, U.K., 2014, pp. 1–14.

[32] P. Barbosa, A. Brito, H. Almeida, and S. Clauss, "Lightweight privacy for smart metering data by adding noise," in *Proc. 29th Annu. ACM Symp. Appl. Comput. (SAC)*, Gyeongju, Korea, 2014, pp. 531–538.

[33] D. Chen, D. Irwin, P. Shenoy, and J. Albrecht, "Combined heat and privacy: Preventing occupancy detection from smart meters," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Budapest, Hungary, 2014, pp. 208–215.

Dong Chen (S'14) received the Bachelor's degree from Xi'an College, Xi'an, China, and the Master's degree from Northeastern University, Shenyang, China, in 2006 and 2010, both in computer science. He is currently pursuing the Ph.D. degree in electrical and computer engineering from the University of Massachusetts at Amherst, Amherst, MA, USA.

He was a Visitor at the School of Computer Science, University of Massachusetts at Amherst, for two years. His current research interests include smart buildings and grids.

Sandeep Kalra (S'13) received the B.E. degree in information technology from Sardar Patel University, Vallabh Vidhyanagar, India, in 2009. He is currently pursuing the M.S. and Ph.D. degrees in computer science from the University of Massachusetts at Amherst, Amherst, MA, USA.

His current research interests include smart grids and connected homes, sustainability, data science, and analytics.

David Irwin (M'07) received the B.S. degree in computer science and mathematics from Vanderbilt University, Nashville, TN, USA, in 2001, and the M.S. and Ph.D. degrees in computer science from Duke University, Durham, NC, USA, in 2005 and 2007, respectively.

He is an Assistant Professor with the Department of Electrical and Computer Engineering, University of Massachusetts at Amherst, Amherst, MA, USA. His current research interests include experimental computing systems with a particular emphasis on sustainability.

Prashant Shenoy (F'13) received the B.Tech. degree in computer science and engineering from the Indian Institute of Technology–Bombay, Mumbai, India, in 1993, and the M.S. and Ph.D. degrees in computer science from the University of Texas at Austin, Austin, TX, USA, in 1994 and 1998, respectively.

He is currently a Professor of Computer Science with the University of Massachusetts at Amherst, Amherst, MA, USA. His current research interests include cloud computing and green computing.

Prof. Shenoy is a Distinguished Member of the Association for Computing Machinery.

Jeannie Albrecht (M'07) received the B.S. degree in mathematics and computer science from Gettysburg College, Gettysburg, PA, USA; the M.S. degree in computer science from Duke University, Durham, NC, USA; and the Ph.D. degree in computer science from the University of California at San Diego, La Jolla, CA, USA, in 2001, 2003, and 2007, respectively.

She is an Associate Professor with the Department of Computer Science, Williams College, Williamstown, MA, USA. Her current research interests include computer systems, including distributed systems, mobile and wide-area networks, operating systems, and green computing.