

Protecting your Daily In-Home Activity Information from a Wireless Snooping Attack

Vijay Srinivasan
University of Virginia
Dept of Computer Science
vs8h@virginia.edu

John Stankovic
University of Virginia
Dept of Computer Science
stankovic@cs.virginia.edu

Kamin Whitehouse
University of Virginia
Dept of Computer Science
whitehouse@cs.virginia.edu

ABSTRACT

In this paper, we first present a new privacy leak in residential wireless ubiquitous computing systems, and then we propose guidelines for designing future systems to prevent this problem. We show that we can observe private activities in the home such as cooking, showering, toileting, and sleeping by eavesdropping on the wireless transmissions of sensors in a home, *even when all of the transmissions are encrypted*. We call this the Fingerprint and Timing-based Snooping (FATS) attack. This attack can already be carried out on millions of homes today, and may become more important as ubiquitous computing environments such as smart homes and assisted living facilities become more prevalent. In this paper, we demonstrate and evaluate the FATS attack on eight different homes containing wireless sensors. We also propose and evaluate a set of privacy preserving design guidelines for future wireless ubiquitous systems and show how these guidelines can be used in a hybrid fashion to prevent against the FATS attack with low implementation costs.

ACM Classification Keywords

C.2.1 [Computer-Communication Networks]: Network Architecture and Design – wireless communication; H.m [Information Systems]: Miscellaneous

General Terms

Security, Design, Experimentation

Author Keywords

Wireless fingerprinting, Side channel privacy attacks, Activity monitoring

INTRODUCTION

Wireless sensors are becoming ubiquitous in homes and residential environments. Elderly monitoring and assisted living facilities are deploying sensors to monitor the medicine cabinet, toilet, shower, sinks, stove, and other appliances [24]. Over 32 million homes in the US have security sensors installed on doors and windows, and motion sensors installed

inside and/or outside the home [21]. Over 5 million homes have X10 devices [6] and ZigBee devices such as wireless doorbells, appliance controls, wireless smoke detectors, and wireless light switches. These sensors are often augmented by personal area networks (PANs) that include wireless pedometers in shoes or wireless EKG sensors on people. These sensors produce a constant record of the *Activities of Daily Living* (ADLs) within a home, thereby enabling a new generation of ubiquitous computing applications such as smart homes, elderly monitoring [24], and home security [6].

ADLs are typically very personal and private, and must be kept secret from third parties. This is particularly true in medical facilities where ADLs are used to infer medical conditions; these facilities are *obliged* by HIPAA regulations [5] to protect this information. In this paper, *we present guidelines for designing wireless ubiquitous computing systems in residential environments to preserve private activity information about the residents*.

In the first half of the paper, we present a powerful new attack that allows us to observe private activities in the home such as cooking, showering, toileting, and sleeping by snooping on the wireless transmissions of sensors in a home, even if all of the transmissions are fully encrypted. This attack needs only the timestamp and the *fingerprint* of each radio message, where a *wireless fingerprint* is a set of features of a RF waveform that are unique to a particular transmitter. Thus, we call this the *Fingerprint And Timing-based Snooping* (FATS) attack. This is a new attack that has not previously been studied or demonstrated. Wireless fingerprinting and traffic timing analysis are both well-studied techniques. However, we are the first to combine these two techniques to create and demonstrate a novel privacy attack in the kind of single-hop, wireless networks common in residential ubiquitous computing.

In the second half of the paper, we propose a suite of techniques to protect against the FATS attack. For example, we hide packet transmissions from the adversary using signal attenuators, we introduce random delays on transmissions, and we generate spurious transmissions in order to decrease the effectiveness of the FATS attack. We find that each of these techniques has a different tradeoff in terms of privacy protection performance, hardware costs, and application/user costs. Based on the cost-benefit profile of each technique, we present guidelines for applying them most effectively in typical ubiquitous computing systems. We evaluate these

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

UbiComp'08, September 21-24, 2008, Seoul, Korea.

Copyright 2008 ACM 978-1-60558-136-1/08/09...\$5.00.

design guidelines and show how many of these guidelines can be used together in a hybrid fashion to yield very high privacy protection with minimal implementation costs.

We empirically evaluate both the FATS attack and our suite of privacy preservation techniques by deploying ubiquitous computing systems in eight different homes. These systems contained sensors on salient objects, including sinks, stoves, showers, and doors, and were used to collect data for a week or more in each home. We tested a diverse range of homes, with different floor layouts, different sets of deployed sensors, and occupied by both single and multiple residents from different age groups and lifestyles. Our results suggest that the FATS attack is highly effective even without knowing any prior information about the home, achieving around 80-95% accuracy on activity recognition in many homes in the best case. The design guidelines that we propose greatly decrease the effectiveness of the FATS attack, reducing the inference accuracy to anywhere from 0 to 15%, while greatly reducing implementation costs.

BACKGROUND AND RELATED WORK

FATS is a type of *side channel attack*, which means that it uses information revealed by a cryptographic system other than the ciphertext to infer either the cryptographic keys or the original data [4]. Some well-known side channel attacks include the TEMPEST attack, which uses leaked electromagnetic radiation from a computer monitor to infer the plain text input to the system [17], and a recent study that exploits physical properties of variable bitrate encoding schemes to infer the movie a person is watching on commercial devices [23].

Wireless fingerprinting is a well-studied technique in which physical characteristics of RF transmissions are used to differentiate between messages from different radios, even when those radios have the same model and manufacturer. Statistical features of transient signal *amplitude* have been used to fingerprint Bluetooth devices [15] with false positive rates of 5% and detection accuracies as high as 93%. Similar results have been observed on 802.11 WiFi radios [14] and on *sensor nodes* using the ChipCon CC1000 radio [22]. We discuss the hardware requirements for fingerprinting when we discuss deployment details in the next section. The above work on *wireless fingerprinting* is different from *software-based fingerprinting*, such as recent work that identifies and tracks users of 802.11 enabled devices by exploiting implicit identifiers in 802.11 network traffic [20]. Wireless fingerprinting is usually used to *enhance* privacy by enabling hardware-based authentication [14], not to compromise it. A recent study proposed using wireless fingerprints to compromise privacy in vehicular sensor networks [11], but did not combine fingerprints with traffic timing analysis.

The FATS attack is different from most existing traffic analysis attacks. Previous work has demonstrated that Internet traffic patterns in wired networks can be used to match a sender with a recipient [10], and multi-hop radio traffic in wireless sensor networks can be used to locate the sensor source or the base station [8, 16]. The countermeasures for

these attacks require changing network flow patterns at the routing level. The FATS attack uses fingerprints to do traffic analysis in a single-hop network, and so is not affected by such countermeasures. Yang et al [9] describe a related traffic analysis attack in which an adversary can infer when a network event has occurred by observing global transmission timestamps alone, but this work does not combine transmission timestamps with wireless fingerprints.

Our multi-tier FATS algorithm infers activities such as cooking, showering and toileting. Activity recognition in the home setting using simple binary sensors is a hard problem and has been well studied in the literature [24, 25]. We are studying a simpler version of this problem here, since our main goal is to show what the adversary can infer using only wireless fingerprints and timing information. Also, the FATS attack assumes a home fitted with simple wireless sensors [2] as opposed to other approaches that use RFID tags on household objects coupled with wearable RFID readers [25]. Our attack is more suited to a *wireless sensor system* than to an *RFID-based system*, since it's easier to snoop on relatively long range wireless transmissions than on short-range RFID signals. We believe that both types of systems will be commonly used in the future.

THE FATS ATTACK

We developed the multi-tier FATS inference algorithm to infer information about a home and its residents from just the timing and fingerprints of radio transmissions. Our inference algorithm is surprisingly robust to the diversity of homes, people and sensed objects in our deployments and can infer detailed resident activity information with high accuracy. We now provide an overview of the FATS inference algorithm. We explain our algorithm in terms of its logical view, as depicted in Figure 1, as well its concrete operations on real data, as depicted in Figure 2. Firstly, figure 1 shows a **Tier 0**, where the adversary only has access to timestamps, but not fingerprints. In Tier 0, the adversary can only detect very general activities such as home occupancy or sleeping.

In **Tier I**, the adversary uses fingerprints to associate each message with a unique transmitter, depicted by the black lines in Figure 2. Then, these transmitters are grouped into *sensor clusters* corresponding to rooms in the home based on similarities in their transmission patterns, shown by the cluster labels on the left of Figure 2. The approximate number of residents in the home can be inferred by observing simultaneous activity in multiple rooms, as shown in Figure 1. In **Tier II**, specific features are first extracted from the combined transmissions of all devices in a spatial cluster, denoted by the red lines in Figure 2 beneath each cluster. These features are passed to a classifier that identifies each room as either a kitchen, bathroom etc. Figure 1 shows that these room labels can be used to infer the number of times the residents visit, for example, the kitchen or bathroom each day. Finally, in **Tier III**, another classifier is used to determine the likelihood of a sensor being the motion sensor, stove sensor etc. This information can be used to recognize subtly different activities, namely cooking hot and cold food or showering, toileting, and grooming, as shown in figure 1.

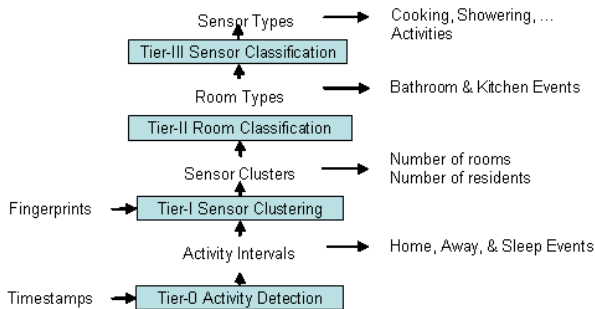


Figure 1. The FATS Inference Algorithm

Other private medical and personal information could be inferred if the adversary is able to make further assumptions, perhaps by knowing information about the specific residents.

In the rest of this section, we describe in more detail, the design and evaluation of the four tiers in our FATS inference algorithm. First, we describe the details of our wireless home deployments on which the FATS attack was demonstrated and evaluated. We then list the evaluation metrics necessary to measure how well the FATS adversary infers private information. Finally, we describe the algorithms used in the four tiers of our inference algorithm. We follow the algorithmic description of each tier with an evaluation of how well that tier infers private information from our system deployments. The evaluation of each tier assumes ideal, unrealistic conditions of 100 % packet reception and 100% fingerprint accuracy. We evaluate our algorithm this way to see what the adversary can learn in the best case. However, we do conclude this section with an evaluation of the overall FATS inference algorithm under practical conditions with both packet loss and fingerprint errors.

Deployment Details

We empirically evaluate the FATS attack by collecting real sensor data from homes containing off-the-shelf wireless X10 motion sensors and contact sensors. Motion sensors were placed in each room and contact switches were placed on doors, sinks, toilets, showers, refrigerators, stoves, and cabinets, some of which are shown in Figure 3. This type of instrumentation might be typical of elderly care or home monitoring applications [19]. Data was collected in each home for seven days or more.

To be sure that our evaluation is not specific to a particular home or type of home, we deployed the X10 devices in eight diverse homes. All homes had different floor plans, a different number of sensors, different items being monitored, and a different number of residents with diverse age groups and occupations. Some homes had three male graduate student residents while other homes had a married couple and mother-in-law. Layouts varied from studios to two-bedroom apartments. A summary of the diversity of homes is shown in table 1. Hereafter, we label the single person homes A through D and the multi-person homes E through H.

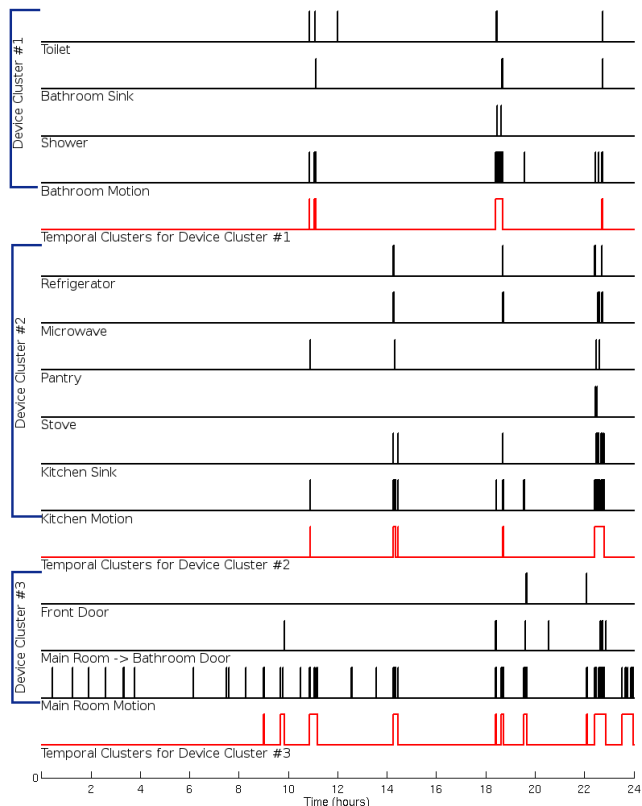


Figure 2. The textual labels and logical groupings are derived from the raw sensor data (black lines) by our tiered inference algorithm.

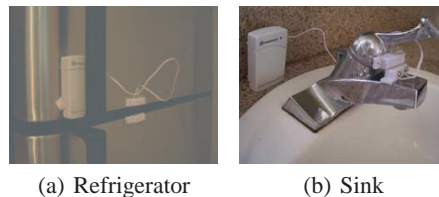


Figure 3. Examples of the sensors we deployed in eight different homes

To fingerprint radio sources, the FATS adversary could either simply use the RSSI values [13] of radio messages, which is less accurate, or use an antenna connected to a high frequency oscilloscope or a commercial integrated signal analyzer with built-in fingerprinting software [1]. Even though many of these snooping devices are currently costly and power hungry, it is expected that cheaper devices will be available as wireless fingerprinting matures and becomes increasingly important for hardware-based authentication. It is easier for an adversary to launch the FATS attack if she has access to uninterrupted power, either from an unattended outdoor power outlet nearby, an adjacent apartment or a distant surveillance area with powerful directional antennae. We have performed experiments to validate that wireless micas inside our office buildings can be snooped upon from the outside. Also, X10 radios have a long radio range and snooping on these devices is as easy as driving around with an X10 receiver to receive unencrypted X10 camera data

Home	Number of Sensors	Number of Rooms	Number of people	Number of firings
A	13	3	1	5888
B	22	5	1	3074
C	16	4	1	4020
D	16	4	1	10326
E	12	4	3	14340
F	15	5	3	20534
G	14	4	2	10964
H	16	5	2	16571

Table 1. Characteristics of the eight homes used in our deployments

[3]. In our deployments, we used an X10 receiver directly inside the home to record a timestamp and device ID for each radio message. We do so in order to study both (i) a best case scenario where the adversary has *perfect* snooping capabilities, and (ii) a more realistic scenario with simulated packet loss and fingerprinting errors.

Evaluation Metrics

Each tier of our inference algorithm will produce a set of resident activity time intervals I that are defined by a start time and an end time. We also produce a set of *ground truth* activity intervals \hat{I} by hand-labeling the sensor data after data collection is over. We use a min-cost bipartite matching algorithm to pair each interval in I with an interval in \hat{I} . Then, we use three metrics to quantify the correctness of the inference algorithm:

1. **Event Detection Rate (EDR):** the percentage of intervals in \hat{I} that were mapped to some interval in I .
2. **True Positive Rate (TPR):** the percentage of intervals in I that were mapped to a real event in \hat{I} .
3. **Duration Accuracy (DA):** the absolute difference in duration between events in I and their matched events in \hat{I} .

A EDR value of 60% would be produced if 10 cooking events occurred but only 6 were detected. A TPR value of 60% would be produced if the adversary detected 10 cooking events of which 4 were false alarms. By measuring both EDR and TPR, we ensure that the FATS algorithm does not achieve a high EDR simply by generating a large number of spurious events. A DA of 60% would be produced if a cooking event takes 100 minutes, and the algorithm indicates an event of either 60 minutes or 140 minutes.

Tier-0: General Activity Detection

An adversary can detect activity in a home even without wireless fingerprinting simply by snooping on radio activity and correlating radio activity with human activity inside the home. In Tier 0, we implemented a simple algorithm that identifies silent periods during the day as *away* events, silent periods during the night as *sleep* events, and all active periods to be *home* events. To reduce the effect of spurious sensor activity, we did not count the first four transmissions each hour as true radio activity. Sleep and home event detection has 85 to 100% duration accuracy (DA) in both

single and multi-person homes, as shown in Figure 5. In multi-person homes, only the aggregate activity can be inferred, such as when *everyone* in the home was sleeping or not. This baseline privacy leak does not require wireless fingerprinting, and can already be applied today to the over 32 million homes in the US that have wireless home security sensors [21]. In subsequent sections, we show that wireless fingerprinting *in combination* with transmission timestamps allows the adversary to infer much more detailed information.

Tier-I Clustering

The goal of Tier I is to identify which sensors are in the same rooms. It does this by assuming that sensors in the same room fire at similar times due to human activity in the room. Thus, we use a *temporal distance* calculated between the transmission patterns of each pair of sensors to cluster together those sensors that have small distances to each other.

We denote the set of all devices (identified using their unique fingerprints) to be ID , and the vector of all transmission timestamps from each device $i \in ID$ to be T_i . We use bracket notation to index into vectors, so the k th timestamp from node i is referred to with $T_i[k]$.

The clustering algorithm is then defined as:

```

forall  $i, j \in ID$ 
  for  $h = 1$  to  $length(T_i)$ 
     $dist_{ij}[h] = \infty$ 
    for  $k = 1$  to  $length(T_j)$ 
       $dist_{hk} = |T_i[h] - T_j[k]|$ 
      if  $dist_{hk} < dist_{ij}[h]$ 
         $dist_{ij}[h] = dist_{hk}$ 
     $D_{ij} = \min(\text{median}(dist_{ij}), \text{median}(dist_{ji}))$ 
 $D' = \text{SPDIST}(D)$ 
 $F = \text{CMDS}(D')$ 
CLUSTER=k-means( $F, k$ )

```

For each pair of devices i and j , we first compute the difference in time between each transmission of i and the *closest* transmission of j , creating a difference vector $dist_{ij}$ with length $|T_i|$. We then calculate the **temporal distance** D_{ij} between devices i and j to be the minimum of the median of the time difference vectors $dist_{ij}$ and $dist_{ji}$. These distances are stored as elements of the symmetric distance matrix D of size $|ID| \times |ID|$, which holds the distances between each pair of transmitters.

Not all devices in the same room will be temporally correlated; for example, the dishwasher may have a large temporal distance from the refrigerator because they are never used together, even if both devices have small temporal distances to something else like the sink or a motion sensor. Therefore, we use Dijkstra's shortest path algorithm to convert the distance matrix D to a new matrix D' of metric distances by replacing each distance D_{ij} by the shortest path distance SP_{ij} through D . We then use classical non-parametric multi-dimensional scaling (CMDS), to convert the distance matrix D' into positions of the $|ID|$ sensors in $|ID|$ -dimensional

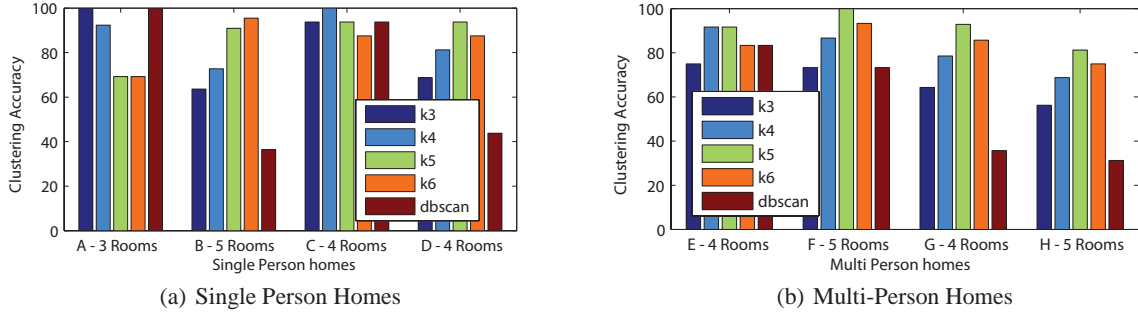


Figure 4. Accuracy of clustering sensors into rooms in each of eight homes, using both *dbscan* and *k*-means with 4 different values of *k*.

space. Finally, we use the *k*-means clustering algorithm to cluster together sensors that are temporally correlated, arriving at a mapping $CLUSTER : ID \rightarrow C$, from the source identity $i \in ID$ of every device to one of the $k = |C|$ clusters. Takada et al [7] describe a related algorithm to group co-located sensors together based on time series data; however, the underlying sensors used, the distance matrix computation and the clustering algorithm used there are different.

Tier-I Evaluation

To compute clustering accuracy, we first compute a maximal min cost bipartite mapping between the set C of computed clusters and the set \hat{C} of true room clusters obtained from the deployment plan. We define a device $i \in ID$ to be *correctly clustered* if i 's computed cluster $c \in C$ is mapped to a room cluster $r \in \hat{C}$ such that device i is actually in room r . We then define **clustering accuracy** to be the proportion of devices that are correctly clustered. Figure 4 shows the clustering accuracy across homes for multiple values of k , and the *non-parametric db-scan clustering algorithm* [12]. Maximal clustering accuracy is achieved when the value of k matches the number of rooms in each home. However, *our overall inference algorithm does not require the exact number k of rooms in the home and is robust to incorrect, larger values k* . The parameterless *db-scan* algorithm performs poorly in several homes where all devices are highly correlated temporally. Thus, we need a parameterized clustering technique like *k*-means with large enough k to enforce a reasonable partitioning of the sensors into clusters corresponding to rooms in the home. It is important to note in figure 4b that clustering accuracy remains high in *multi-person homes* in spite of simultaneous activity in different rooms; this is because we still get sensors firing in the same room most of the time. Though we do not show it here, it is possible to infer the number of residents currently in the home by tracking simultaneous activity in multiple rooms.

Tier-II Room Classification

The goal of Tier II is to identify the function of each room as a bathroom, kitchen, bedroom, or living room. This tier makes two assumptions: (i) different houses have similar rooms (ii) similar rooms across homes can be identified using specific features of room usage. To identify the function of each room, this tier passes features computed for each

room based on the the overall sensor transmissions from the room to a bi-partite matching based classifier.

Once the devices are clustered, we generate an overall series of timestamps T_c of all transmission timestamps from all sensors in sensor cluster $c \in C$. We also generate **temporal activity clusters**, used in the features below, by using the *db-scan* algorithm to cluster the timestamps in T_c . Each temporal activity cluster forms a continuous temporal block from T_c with a relatively high density of sensor firings. *db-scan* [12] performs well here because it automatically leaves out outliers and computes high-density clusters unlike *k*-means. We then generate a number of features for every room cluster c from the series T_c listed below.

- the number of transmissions per day from the room
- the median inter-transmissions time within a room
- the median length of temporal activity clusters
- the total number of transmissions during the day
- the total number of transmissions during the night
- the cluster to transmit first after long silence periods
- a histogram of transmission, with four hour granularity

We use these features to create a feature vector F_r for every room r . To classify rooms in a test home, we set apart a small number of other homes from our deployment to provide *training data*. We define R to be all possible room labels (e.g., bedroom, kitchen, bathroom, etc) and create a single feature vector F_r for every label $r \in R$ by averaging the feature vectors of all rooms in the training data with the same label. Then, to label the rooms in a test home, we compute a *min-cost bipartite* matching between the feature vectors $F_c : c \in C$ from the clusters of that home and the training feature vectors $F_r : r \in R$. We define the cost of a match between two feature vectors to be the sum of the *Euclidean distances* between corresponding individual features.

The resulting bipartite matching represents the room labeling. Unlike a conventional classifier, the matching process allows us to enforce mutual exclusivity of room labelings; for e.g., a home with three rooms cannot have two bathrooms or three kitchens. When necessary, we can allow for multiple rooms with the same label in a home by simply including extra copies that room in the set of room labels. In our experiments, we observed that all rooms were correctly labeled across both single and multi-person homes. In this tier, we infer the timing and duration of *bathroom and kitchen visits*

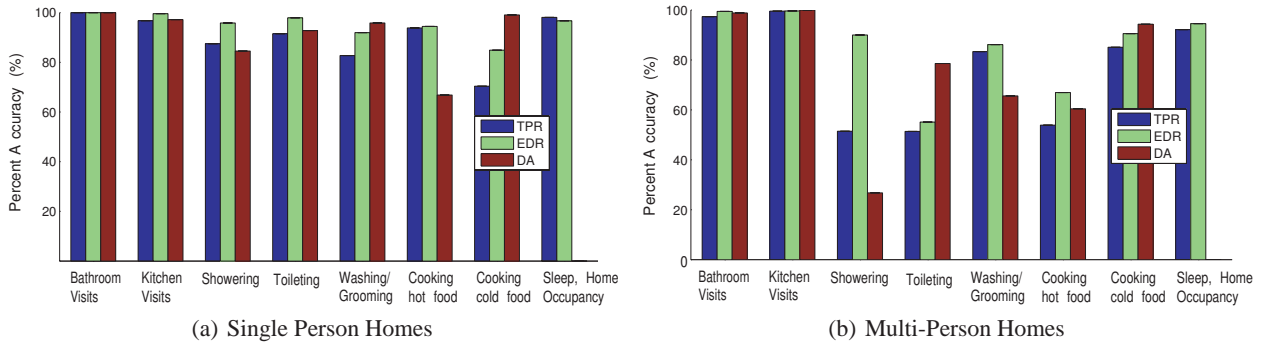


Figure 5. Overall Inference Accuracy on events detected in Tier 0, Tier II, and Tier III, using all three metrics: EDR, TPR, and DA.

by simply counting the temporal activity clusters that occur in each room classified as the bathroom or kitchen.

Tier-II Evaluation

Assuming a best case scenario of perfect packet reception and fingerprinting accuracy, figure 5 shows that the adversary can infer the duration and timing of bathroom and kitchen visits with 95-100% accuracy across both single and multi-person homes using our room classification algorithm. Of course, in multi-person homes, the variables represent how many times *all* residents in the home visited the bathroom or kitchen in total.

Tier-III Sensor Classification

The goal of Tier III is to identify activities in the home, such as cooking, showering, or toileting. This is a two step process. In step 1, we classify sensors to obtain a **mapping vector** for each unknown sensor indicating the likelihood of its matching with known sensors such as the stove, shower etc. The mapping vector M_s for sensor s is indexed by the known sensor types; for example, $M_s[stove]$ indicates the probability that sensor s is the *stove* sensor. In step 2, we use these mapping vectors to classify activities based on which sensors are likely to be active during an activity. In this tier, we assume that we can use specific features in sensor firing patterns observed across homes to obtain a mapping vector for each sensor. However, we do not require the sensor classification to be 100% accurate and show here how inaccurate mapping vectors can be used to accurately classify activities.

We calculate a feature vector for each sensor, using features similar to those described in Tier II, obtained from temporal characteristics of sensor firings. Due to space constraints, we don't discuss the exact features used here. To classify each sensor, we pass each feature vector through a standard *linear discriminant analysis* (LDA) classifier. For each of the eight homes, we constructed this classifier using training data obtained from a subset of the remaining seven homes with hand-labeled sensor types. We do not assume that all houses have exactly the same set of sensors, but we do assume that all types of sensors in the test home have been observed at least once in a training home. We train a separate classifier for each room in a home: a sensor from the kitchen cluster in Tier II will not be classified as a bathroom sensor. Thus, the room classification results from Tier II are used to

improve the sensor classification results in Tier III. The output of this classification procedure is the mapping vector M_s for each sensor s in a home. In our experiments, the mapping vectors were accurate for bathroom sensors, but were not very accurate for kitchen sensors, where activities like cooking produced similar features in several sensors, such as the stove and cabinet sensors. These objects are often either misclassified as each other or partially classified as multiple objects. However, these incorrect classifications can still be used to recognize activities in Tier III.

To recognize activities, we calculate a feature vector of every *temporal activity cluster* in every device cluster. These feature vectors include (i) start time, (ii) duration, and (iii) the total number of times that each known sensor type transmits. Feature (iii) is obtained by adding the mapping vectors for every sensor firing in the activity cluster. If a device is partially classified as multiple types of sensors, partial counts are maintained in the feature vector. This ensures that sensors used in the same activity (such as the pantry and stove used in cooking) that are misclassified or partially classified as each other do not affect the overall counts of known sensor firings in feature (iii). We then classify each *temporal activity cluster* as an activity using an LDA classifier that was trained on other homes with hand-labeled activity labels. We used this approach to recognize *showering, washing/grooming, and toileting* activities in the bathroom, and to recognize *cooking hot food* and *preparing cold food* in the kitchen.

Tier-III Evaluation

Figure 5 shows the activity recognition results for Tier III. In single person homes, the average accuracy of the adversary in inferring detailed activities is around 80%, while the accuracy in multi-person homes is lower, but still well above the baseline of random guessing. The lower accuracy in multi-person homes is due to spatial clustering errors introduced by simultaneous activity in many rooms; for example, in home E, an incorrect clustering of the shower sensor with the living room prevented us from detecting showering events. We note, however, that the high proportion of activities that are correctly classified indicate that the simple assumptions made by the FATS attack in Tiers I, II, and III appear to be true across our diverse sample set of homes, and that this attack can be used to infer private information about the res-

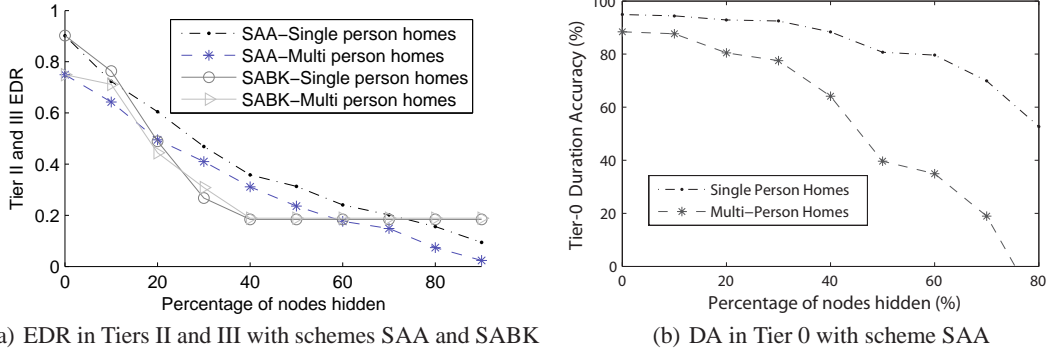


Figure 6. Effect of Signal Attenuators on event detection rate (EDR) of Tier II and III events, and on duration accuracy (DA) of Tier 0 events

idents of a home simply by snooping on the radio messages of its sensors.

FATS attack evaluation under realistic conditions

We now evaluate the FATS attack in practical, non-ideal settings, where the adversary will observe both packet loss and fingerprint errors. The evaluation in figure 6 models packet loss as a percentage of the nodes hidden from the adversary. Even under pessimistic scenarios such as 30% of the nodes hidden from the adversary, the FATS attack still infers about 50% of Tier II and Tier III events and detects Tier 0 events with about 80% DA. Tier 0 DA is unaffected by fingerprint errors, since it simply relies on the transmission timestamps. Figure 10 shows how fingerprint errors, modelled as a feature deviation in the x-axis and explained in detail in the next section, affect Tier II and Tier III EDR. Under pessimistic feature deviations of about 50% of the total feature space (deviations of 20 for a feature space of 40), the adversary still infers upto 50% of Tier II and Tier III events in the home. Assuming both 20% packet loss and 20% feature deviations, the adversary still infers Tier II and Tier III events with 40% EDR, and Tier 0 events with 85% DA on average.

DESIGN GUIDELINES TO ENHANCE PRIVACY

In this section, we evaluate a number of techniques to thwart the FATS attack. Each of these techniques has a different cost-benefit profile, and we analyze how and when each technique is most effective. We present the following five guidelines for building wireless ubiquitous computing systems in homes or residential environments. We justify each guideline in subsequent subsections. We conclude by presenting a **hybrid approach** that combines many of these guidelines to achieve very high privacy protection with very low implementation costs.

1. **Signal attenuators** should be deployed in a select few rooms such as kitchens or bathrooms where many activities occur to effectively mask activities in these rooms.
2. **Random delays** of the order of 15 to 20 minutes should be added to the transmissions of sensors in the bathroom and kitchen that are involved in short duration activities to effectively hide these activities.
3. **Periodic transmissions** should be used on binary or low bandwidth sensors that are typically involved in long-duration

activities, such as bedroom and living room sensors.

4. **Fingerprint masking** should be used on time-critical sensors like fall detection sensors, where latency introduced by random delays or periodic transmissions is unacceptable, or on sensors/rooms where signal attenuators are infeasible.
5. **Spurious or fake transmissions** should be combined with real transmissions for sensors such as camera or microphone sensors that cannot afford the high energy cost from periodic transmissions.

Using Signal Attenuators

The most obvious approach to protect against the FATS attack is to prevent the adversary from hearing messages in the first place. There are several possible signal attenuators, and we list three here: (i) Using very low power transmissions and a multi-hop route to the base station: this incurs a moderate hardware cost in terms of additional router nodes, and reduces reliability (ii) Using a wired connection to the base station: this requires quite a lot of deployment effort and time, and (iii) Using Faraday cages: this is expensive to set up and prevents outside communication from hidden rooms, which is essential for eldercare applications. We recommend scheme (i) since it has the least cost among the various schemes, though it does not have the same protection guarantees as scheme (ii).

We implement two schemes for signal attenuators in our evaluation, namely: (i) **SAA** - Signal Attenuators in All rooms, and (ii) **SABK** - Signal Attenuators in the Bathroom and Kitchen only, as per design guideline #1. Figure 6 shows the effectiveness of the FATS attack as we increase the percentage of nodes hidden by signal attenuators from 0-90%. Figure 6(a) shows that signal attenuators are very effective at reducing the EDR for Tier II and Tier III activities; with 40% hidden nodes, EDR is reduced to about 30-35% in scheme SAA and to 20% in scheme SABK. Thus, many of the activities detected in Tiers II and III can be hidden effectively by using signal attenuators in just the bathroom and kitchen. Figure 8 shows that signal attenuators do not have a strong effect in reducing the Tier II and Tier III DA and TPR. The TPR actually increases as more nodes are hidden, because the number of events detected becomes small enough that almost no events are spurious. Figure 6(b) shows that signal

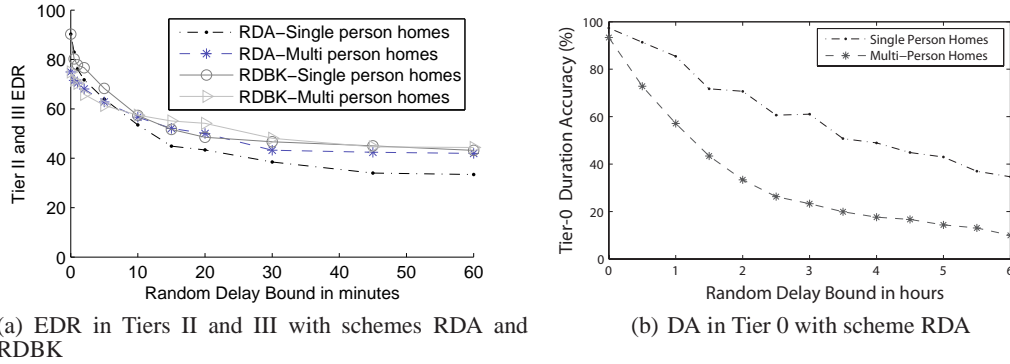


Figure 7. Effect of random delays on event detection rate (EDR) of Tier II and III events, and on duration accuracy (DA) of Tier 0 events

attenuators, even if implemented in the entire home (SAA), are not very effective at reducing the DA for Tier 0 information; with 40% hidden nodes, the DA is nearly 90% in single person homes, and is over 60% in multi-person homes. This is because the sensors that are not hidden by the signal attenuators still reveal the *presence* of human activity very well. Thus, we recommend in design guideline #1 that signal attenuators be used in select rooms such as the bathroom and kitchen (SABK) where many activities occur to reduce implementation cost and hide these activities effectively.

Using Random Delays

Because all tiers of the FATS algorithm rely on transmission timestamps, modifying the transmission time is one way of reducing the effectiveness of the attack. Thus, we propose adding random delays to sensor transmissions bounded by a maximum delay parameter. We implemented two schemes in our evaluation of random delays: (i) **RDA** - Random Delays on All sensors, and (ii) **RDBK** - Random Delays on Bathroom and Kitchen sensors involved in short duration activities only, such as cooking and toileting. Figure 7 shows that the effectiveness of the FATS attack decreases as increasingly long random delays are added to sensor transmissions. Figure 7(a) shows that even small random delays of about 10 minutes are highly effective at reducing the EDR for Tier III events such as cooking and showering to about 50-60% under both schemes RDA and RDBK. This is because random delays introduce errors in Tiers I, II and III which use the device transmission timestamps as input. Also, figure 8 shows that random delays of about 30 minutes reduces both DA and TPR of Tier II and III events to about 40%; i.e., more than half the events detected by the adversary are false positives, and the inferred duration of these events is highly inaccurate. As seen in figure 7(b), longer random delays of about 3 hours reduce Tier 0 DA to about 60% in single person homes and a much lower 25% in multi-person homes. In multi-person homes, there are only a few hours per day with no activity from *all* residents, and 3 hour random delays ensure that almost no period is inactive, resulting in the larger duration errors. However, we do not recommend such long delays of the order of hours to hide long duration Tier 0 activities such as home presence and sleeping.

Despite their effectiveness, random delays conflict with the

requirements of real-time sensors such as wireless light switches, or fall detection sensors that need to transmit data immediately. Also, we need to consider if random delays of the order of 15-30 minutes are acceptable to the end users; they might be certainly acceptable to remote healthcare providers who are only interested in long term trends such as a decline in the ability of residents to perform ADLs. For other users, we need user queries to verify if such delays are acceptable. Given the similar performance of schemes RDA and RDBK in protecting Tier III events, we recommend in design guideline #2 that random delays of the order of 15 minutes be applied to non-emergency sensors in the bathroom and kitchen if the end users find such delays acceptable.

Using Periodic Transmission

If a sensor transmits periodically instead of only when it has data to transmit, it makes the transmission timestamps independent of the data, making it impossible for the adversary to infer any information. Thus, periodic transmission on all sensors guarantees 100% privacy. We first estimate the extra power consumed by periodic transmissions for typical *binary home sensors*, using empirical power consumption data on the telos mote with the CC2420 radio [18], a popular hardware platform in wireless sensor network research. We assume that a mote sleeps for a latency period P and wakes up to transmit a large enough data payload to capture *binary event information* for the past P seconds, along with the mandatory header fields. The total percent reduction in node lifetime for different periods P is shown in Figure 9. As the period P increases, the percent reduction in node lifetime decreases along with the power consumed by periodic transmissions; this is because as the period increases, the number of packets and the associated energy wastage from packet overhead decreases. We note that *the total reduction in node lifetime for periodic transmission with a period of 8 seconds is only 8.75% of the total original lifetime of the node*; thus, periodic transmissions is an excellent solution for binary sensors. The low power consumption here is because the network is one-hop, and nodes do not need to go into receive mode, unlike other sensor network deployments.

Despite their effectiveness, periodic transmissions cannot be applied to (i) real-time sensors, because of the delay limitation, and (ii) high bandwidth sensors, because of the exces-

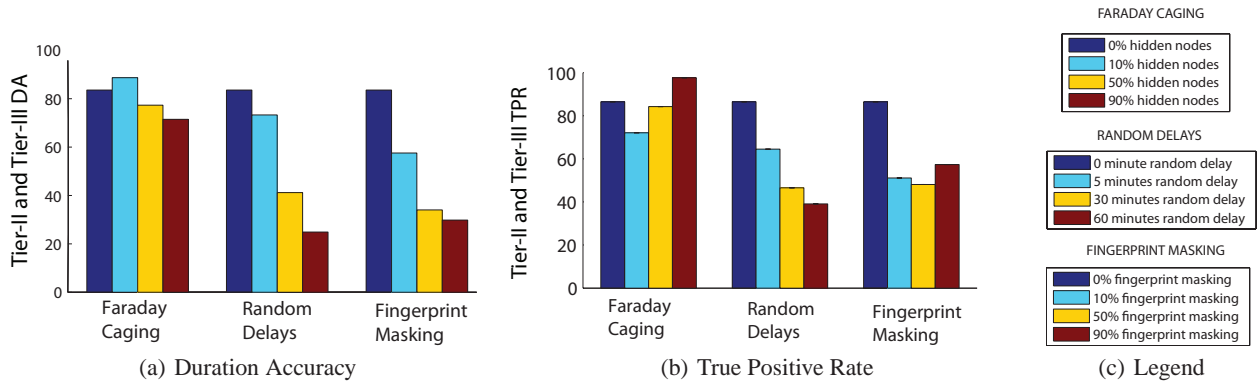


Figure 8. Effect of signal attenuators, delays, and fingerprint masking on duration accuracy (DA) and true positive rate (TPR) of Tier II and III events

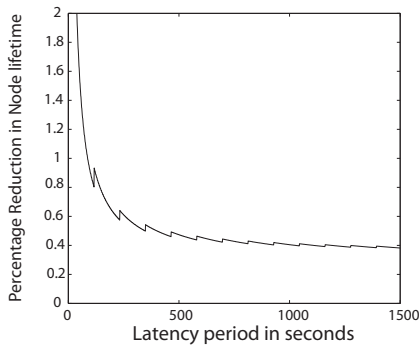


Figure 9. Effect of period of transmission on node lifetime

sive power consumed by redundant transmissions. Thus, we recommend in design guideline #3 that periodic transmissions be used for binary, non-emergency sensors in the living room and bedroom. This will hide Tier 0 DA with only a negligible power cost, where the other techniques are expensive. Indeed, we implemented periodic transmissions on bedroom and living room sensors and found that this reduced Tier 0 DA by 17% in single person homes and a significant 71% in multi-person homes.

Using Fingerprint Masking

We can also preserve privacy through *fingerprint masking*, in which we hide the fingerprint of a transmitter. One approach to doing this might be to use potentiometers instead of resistors for all radio circuitry, and to vary these during each transmission. Another approach might be to wire together multiple sensors and use a common radio for all of them, so that the individual source fingerprint is hidden. As a variation, a sensor can also be wired to multiple radios, each of which might have a different set of sensors assigned to it, to further obfuscate the fingerprints. One problem with fingerprint masking is that it creates an *arms race* scenario in which the adversary and hardware designer must continually try to outsmart each other to uncover and hide new features respectively. Because of this challenge and the need to change existing radio hardware, we recommend in design

guideline #4 to use fingerprint masking only in sensors that cannot tolerate any delays, such as wireless light switches or fall sensors and in sensors/rooms where signal attenuators are infeasible.

For evaluation purposes, we use a simple model to simulate various degrees of fingerprinting error. We assign a scalar fingerprint ID_i using a uniform random distribution to each device i such that $0 < ID_i < L$. When a device transmits, the adversary observes some fingerprint $\hat{ID} = \mathcal{N}(ID_i, \sigma)$ and identifies the transmitter to be $\text{argmin}_j |\hat{ID} - ID_j|$. Thus, fingerprinting errors are likely to increase as the standard deviation σ increases. We simulate fingerprint errors on the raw data by gradually increasing σ until it equals the actual length of the feature space L , set to 40 in our case. Figure 10 shows that small fingerprint errors are effective at reducing Tier II and III EDR. Also, figure 8 shows that fingerprint masking has a similar effect to random delays in terms of Tier II and III DA and TPR, causing a significant drop in both duration accuracy and true positive rate. Introducing fingerprint errors is effective because it distorts sensor clustering as devices from different rooms appear to fire together, and also distorts the features used in our classifiers.

Introducing Spurious or Fake Transmissions

Yang et al [9] propose a countermeasure for traffic timing analysis in which fake and real transmissions are combined in such a way that a fixed probability distribution is maintained for time between transmissions. Real transmissions are delayed to follow the probability distribution when necessary. This countermeasure can be applied to our FATS attack too. Similar to periodic transmissions, fake transmissions would essentially ensure 100% privacy with some transmission delay but with a lower power consumption, since we are only adding some fake packets rather than transmitting large constant data payloads periodically. Thus, we recommend in design guideline #5 to use spurious or fake transmissions on high bandwidth sensors such as cameras or microphones that transmit data occasionally.

Hybrid Schemes

Based on our design guidelines, it is clear that each of our privacy solutions is best suited to certain kinds of sensors and

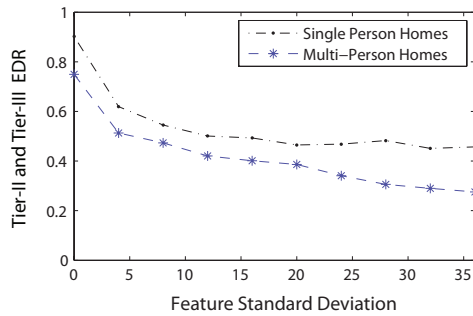


Figure 10. Effect of fingerprint masking on event detection rate (EDR) of Tier II and III events

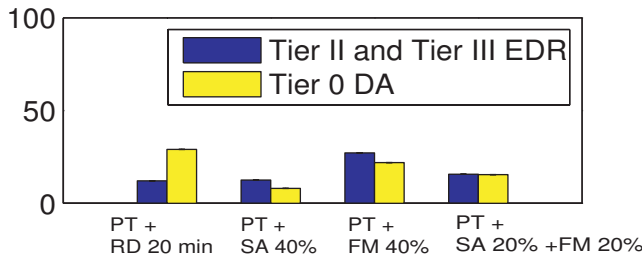


Figure 11. Effect of hybrid schemes on Tier II and Tier III EDR and Tier 0 DA

is better at protecting certain types of information. Therefore, many of these solutions will be used in *combination* in a real home wireless system with diverse sensor types. To demonstrate that such a hybrid approach offers excellent privacy protection, we implemented several hybrid solutions that include the techniques listed in our design guidelines, namely PT - Periodic Transmissions on living and bedroom sensors, Signal Attenuators and Random Delays (SA and RD) on bathroom and kitchen sensors, and FM - Fingerprint Masking. The effect of these hybrid schemes on Tier II and Tier III EDR and Tier 0 DA is shown in figure 11. The hybrid schemes shown are annotated with details relating to the extent to which each technique was applied: for example, *40% FM* refers to 40% feature deviations, *RD 20 min* to a 20 minute random delay, and *40% SA* to 40% hidden nodes. Based on the results shown in figure 11, scheme *PT + RD 20 min* should be used effectively in homes where such delays are acceptable to hide information from all tiers of the FATS attack with minimal costs. For homes where such delays are not acceptable, one of the other hybrid schemes should be chosen based on the implementation costs affordable. Scheme *PT + 20% SA + 20% FM* looks promising, since it requires neither extensive signal attenuation nor extensive fingerprint masking, but achieves excellent privacy preservation with relatively small costs.

CONCLUSIONS

Our design guidelines to guard against the FATS attack may become increasingly important as wireless sensors become more ubiquitous in homes and residential environments. Millions of homes are already vulnerable to the FATS attack, and new systems are being deployed at an ever increasing rate. Also, we believe that the FATS attack is just one in-

stance of many potential physical-layer privacy attacks on wireless ubiquitous systems. Other attacks could be carried out in offices, factories, and even in urban-scale wireless networks. For example, a company that shares an office building with a competitor may infer a new product launch by the competitor by observing increased traffic in certain areas. This study demonstrates the power and ease of physical-layer wireless privacy attacks such as FATS, and our design guidelines are a first step toward thwarting such attacks.

REFERENCES

1. Anritsu high performance signal analyser. http://www.scs.carleton.ca/~jhall2/Publications/anritsu_us.pdf.
2. Assisted living and residential monitoring network project. University of Virginia ALARMNET project, <http://www.cs.virginia.edu/wsn/medical/>.
3. Dark deal hacking wireless video cameras. <http://www.g4tv.com/techtv/vault/features/46880/>.
4. Hagai Bar El, *Introduction to Side Channel Attacks*, <http://www.hbarel.com/publications.htm>.
5. United States department of health and human services, HIPAA regulations and standards. <http://www.hhs.gov/ocr/hipaa/>.
6. X10 home security home automation electronics. <http://www.x10.com>.
7. Toshihiro Takada et al, Proximity mining: Finding proximity using sensor data history. In *WMCSA*, 2003.
8. Jing Deng Han et al, Countermeasures against traffic analysis attacks in wireless sensor networks. In *SecureComm*, 2005.
9. Yi Yang et al, Towards event source unobservability with minimum network traffic in sensor networks. In *WiSec*, 2008.
10. D. Chaum. Untraceable electronic mail, return addresses and digital pseudonyms. In *Communications of the ACM*, pages 84–88, 1981.
11. F. Dotzer. Privacy issues in vanet. In *workshop on Privacy Enhanced Technology*, 2005.
12. M. Ester, Kriegl, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *International Conference on Knowledge Discovery and Data Mining*, 1996.
13. D. B. Faria and D. R. Cheriton. Detecting identity-based attacks in wireless networks using signalprints. In *Wise*, 2006.
14. J. Hall, M. Barbeau, and E. Kranakis. Enhancing intrusion detection in wireless networks using radio frequency fingerprinting. In *Communications Internet and Information Technology*, November 2004.
15. J. Hall, M. Barbeau, and E. Kranakis. Detecting rogue devices in bluetooth networks using radio frequency fingerprinting. In *IASTED International Conference on Communications and Computer Networks*, October 2006.
16. P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk. Enhancing source location privacy in sensor network routing. In *Int Conference on Distributed Computing Systems*, 2005.
17. M. Kuhn. Electromagnetic eavesdropping risks of flat-panel displays. In *Workshop on Privacy Enhancing Technologies*, 2004.
18. R. Lim. Wireless fire sensor network demonstrator. Master's thesis, ETH Zurich, 2006.
19. B. Logan, J. Healey, M. Philipose, E. M. Tapia, and S. Intille. A long-term evaluation of sensing modalities for activity recognition. In *Ubicomp*, 2007.
20. J. Pang, B. Greenstein, R. Gummadi, S. Seshan, and D. Wetherall. 802.11 user fingerprinting. In *MobiCom*, 2007.
21. Parks Associates Research and Analysis for Digital Living. Home security system forecasts: 2005 and beyond, November 2005. <http://www.parksassociates.com/research/reports/tocs/2005/security.htm>.
22. K. B. Rasmussen and S. Capkun. Implications of Radio Fingerprinting on the Security of Sensor Networks. Technical Report 536, ETH Zrich IFW, 2006.
23. S. Saponas, J. Lester, C. Hartung, S. Agarwal, and T. Kohno. Devices that tell on you privacy trends in consumer ubiquitous computing. In *Usenix Security Symposium*, 2007.
24. E. M. Tapia, S. S. Intille, and K. Larson. Activity recognition in the home setting using simple and ubiquitous sensors. In *Proceedings of PERSASIVE*, 2004.
25. D. Wyatt, M. Philipose, and T. Choudhury. Unsupervised activity recognition using automatically mined common sense. In *AAAI*, 2005.