Countering Presence Privacy Attack in Efficient AMI Networks Using Interactive Deep-Learning

Mohamed I. Ibrahem^{*1}, Mahmoud M. Badr^{†2}, Mohamed Mahmoud^{†3}, Mostafa M. Fouda^{‡4}, and Waleed Alasmary^{§5}
*Department of Cyber Security Engineering, George Mason University, Fairfax, VA, USA
†Department of Electrical and Computer Engineering, Tennessee Tech University, Cookeville, TN, USA
‡Department of Electrical and Computer Engineering, Idaho State University, Pocatello, ID, USA
§Department of Computer Engineering, Umm Al-Qura University, Makkah, Saudi Arabia
Emails: ^{1}mibrahem@gmu.edu, {^{2}mmbadr42, ^{3}mmahmoud}@tntech.edu, ^{4}mfouda@ieee.org, ^{5}wsasmary@uqu.edu.sa

***Abstract*—Reporting fine-grained power consumption readings periodically in advanced metering infrastructure (AMI) results in transmitting a massive amount of data by each smart meter (SM). To collect these readings efficiently, change and transmit (CAT) approach can be used. In CAT, the SM sends a consumption reading only when there is enough change in the consumption, which reduces the number of transmitted readings. However, using the CAT approach may trigger attackers to launch a presence-privacy attack (PPA) to infer sensitive information such as the absence of the house occupants by analyzing their SM's transmission pattern. Therefore, in this paper, we propose a scheme, called "STID", for collecting the power consumption readings efficiently in AMI networks while preserving the consumers' privacy by transmitting spoofing transmissions based on an interactive deep-learning defense model. First, we create a dataset that contains the CAT transmission patterns using real power consumption readings and a clustering technique. Next, we train a deep-learning-based attacker model to launch PPA, and the results indicate that the success rate of the attacker is about 90%. Finally, to mitigate the PPA, we train a defense model using deep-learning to transmit spoofing transmissions. The evaluations of our envisioned STID scheme demonstrate a significant reduction in the attacker's success rate while achieving high efficiency in terms of the number of readings that should be transmitted. Our measurements indicate that our proposed STID can reduce the attacker's success rate to 6.12% and increase efficiency by about 38% compared to transmitting readings periodically.**

***Keywords*—Smart grid, privacy preservation, traffic analysis attack, and AMI networks**

I. Introduction

Smart grid is a revolutionary version of the traditional power grid which targets reducing the greenhouse gas emissions and making the electricity delivery more reliable by using more renewable energy resources [1]. Advanced metering infrastructure (AMI) is an important part in the smart grid, which enables two-way communication between the smart meters (SMs) deployed at consumer premises and the system operator (SO). This communication facilitates the periodic collection of the fine-grained power consumption readings sent by SMs (e.g., every few minutes) for load monitoring and energy management [2], [3].

978-0-7381-1316-6/21/$31.00 ©2021 IEEE

However, the periodic transmission of the fine-grained readings leads to inefficient use of the available resources (bandwidth) because a tremendous amount of data should be transmitted by each SM. This problem is getting aggravated since the AMI networks have millions of SMs and it is a cost prohibitive to transmit this large amount of data, especially when using cellular networks. For efficient collection of the power consumption readings, an approach, called change and transmit (CAT), can be employed. In CAT approach, there is no need to send the consumption reading if no enough change occurs in the consumption [4]. To be more precise, the SM sends a reading when the current consumption is greater or less than the last reported reading by more than a predefined threshold.

On the other hand, using the CAT approach causes inevitably a privacy problem in which attackers can launch a presence-privacy attack (PPA) to infer sensitive information on the house occupants by analyzing their SM's transmission pattern. For example, the inferred information can reveal the absence of the occupants, their sleeping times, the number of occupants, etc [5], because these events have a direct influence on the transmission pattern. This is a good news to the community of thieves to find the right time to commit crimes, e.g., kidnapping, pillage, and burglary, using these private information [5]. Encrypting the consumption readings cannot solve this problem because attackers analyze the transmission patterns without the need to decrypt the readings [4]. What makes this problem serious is that the attack is not easy to be detected because it is launched by only intercepting and capturing the transmission patterns to analyze them without affecting the communications. Furthermore, the broadcast nature of the wireless communication that is usually used in AMIs makes the collection of the victim consumer's transmission patterns easy [4].

Therefore, in this paper, we address the research problem of *how to thwart PPA, while achieving a high efficiency in terms of the number of readings that should be transmitted.* We also formally present an adversarial model and envision efficient scheme for collection of power consumption readings, referred to as STID, while thwarting PPA using interactive deep-learning. Our contributions in this work can be summarized as follows.

- To the best of our knowledge, this paper is the first to investigate using deep-learning to launch PPA. Comparing to the literature, the proposed attack model is more sophisticated, which is trained on the transmission patterns of the consumers to infer the absence of the house occupants.
- A hybrid interactive deep-learning-based defense model is proposed to mitigate PPA by generating patterns that look like the patterns transmitted when the occupants are present at home. Our proposed defense achieves a reasonably low attacker's success rate, i.e., around 10%, compared to more than 60% using the proposed scheme in [4].
- The proposed defense model can be used for all consumers including the new ones who do not have a history of transmission patterns.
- We propose an approach for labeling the dataset to be used in training and evaluating the proposed attack and defense models.
- The evaluations of our scheme indicate that our scheme can increase the efficiency by reducing the number of transmitted readings by 38% compared to sending consumption readings periodically.

The remainder of this paper is organized as follows. The related works are discussed in Section II. Then, our considered system models are delineated in Section III. Section IV presents the preliminaries used in our work. Next, Section V presents the dataset created for training our models. The proposed attack and defense models are discussed in Section VI. The performance of our scheme is evaluated in Section VII. Finally, the conclusion is drawn in Section VIII.

II. Related Work

Most of the existing schemes that collect the power consumption readings while preserving the consumers' privacy [5]–[8] consider only periodic transmission of the fine-grained readings in AMI networks. These schemes preserve consumers' privacy by using encryption techniques to hide the consumers' power consumption readings, but as mentioned earlier, using encryption techniques is not enough to preserve privacy in case of using the CAT approach. As a result, we focus in this paper on hiding the information that can be inferred from analyzing the consumers' transmission patterns to preserve their privacy.

Few research works attempt to study the use of CAT approach in smart grid in different concerns [4], [9]–[12]. For instance, some works investigated using the CAT approach in different applications such as demand/response [9] and load disaggregation [10], while other works [11], [12] paid attention to find a good threshold value for the consumption change that triggers transmitting a reading. However, none of these research works consider consumers' privacy.

PPA, that can be launched in case of using CAT approach, has been investigated by Li *et. al.* [4]. A defense scheme is proposed to mitigate PPA by sending artificial spoofing transmission (ASP). Two heuristic-based methods are used in ASP to generate the spoofing transmissions, including the Poisson packet generation and history template-based generation. In Poisson packet generation method, when the house occupants are present at home, the average number of power consumption changes during different time periods are stored in the SM's memory. This number of changes can be used when the occupants leave to generate spoofing transmissions using the corresponding number of changes stored in the memory. On the other hand, in the history template-based generation method, the generation of the spoofing transmissions uses old transmission patterns such that they can be repeated or the spoofing transmissions are randomly generated according to the distribution of the time periods between the power consumption changes.

Nevertheless, ASP suffers from significantly high attacker's success rate (above 60%). Assuming that the attacker has old transmission patterns, she uses a K-S test to calculate the error between the distribution of the collected SM's transmission pattern with the distribution of the patterns she has. The resulting error can indicate how the distributions are close to each other, i.e., if the error between them is small enough, it can be concluded that the transmissions are spoofing, and hence, there is no occupants in the house. Moreover, the proposed attack mechanism is very naive, i.e., the attacker analyzes the collected transmission patterns from SMs by using the K-S test. As a result, a more sophisticated attack mechanism may lead to a higher success rate. In addition, the transmission patterns of only one household over only 33 days are considered, and this is a very limited study. Finally, the proposed scheme is not general, i.e., it can be used for only one consumer, because the generation of the spoofing transmission depends on the distribution of the transmission patterns of each consumer. Hence, in this paper, we address these limitations by proposing a new scheme that is based on a deep-learning approach which usually performs better than heuristic-based approaches.

On the other hand, the insertion of spoofing transmissions has been investigated in wireless sensor networks (WSNs) to counter traffic analysis attacks [13], [14], into which PPA may be categorized. The proposed schemes in [13], [14] cannot be used in AMI to counter PPA since WSNs have different characteristics and privacy problems as follows. First, the objective of the proposed schemes is to hide the information about the location of the source/destination sensor nodes, while our target is to preserve the privacy of the consumers' activities in AMI networks. Second, the energy consumption in WSNs is restricted because the sensor nodes are powered using a battery, but this restriction does not exist in AMI networks.

III. System Models

A. Network Model

Our considered network model, as shown in Fig. 1, includes a number of SMs and SO. SMs are deployed at the consumers' premises to send the real-time power consumption readings only when the current consumption is greater or less than the last reported reading by more than a predefined threshold. For

This is the author's version of an article that has been published in IEEE.
The final version of record is available at https://doi.org/10.1109/ISNCC52172.2021.9615798

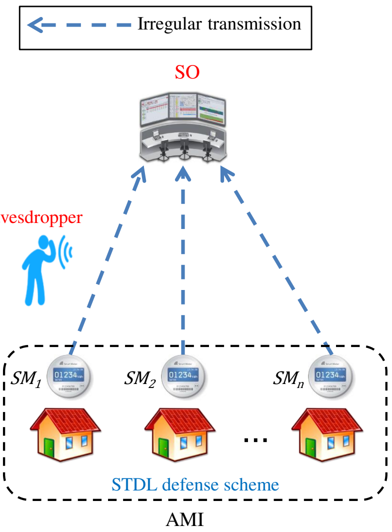


Fig. 1: Network Model.

example, at 3% threshold, the SM should report a reading when the change in the current consumption comparing to the last reported consumption is above +3% or below -3%. The house occupants should trigger the STID defense scheme when he/she leaves home so that it decides interactively whether the SM needs to send a spoofing transmission to the SO at the current time slot to countermeasure PPA.

B. Threat Model

PPA may be launched by internal or external attackers. Internal attackers, e.g., the SMs or the SO, are considered honest-but-curious, i.e., they do not disrupt the communication and follow the protocol, but they are curious to learn the absence of the house occupants. Moreover, an external adversary, e.g., eavesdropper, as shown in Fig. 1, can capture and collect all the exchanged transmissions between the SMs and the SO over a time interval. These collected transmission can help the attacker to construct the transmission patterns of the consumers to learn whether they are on travel or present at home by analyzing their transmission patterns. This is due to the broadcast nature of the wireless communication that is used in AMI networks which facilitates intercepting signals [4]. Also, it is assumed that the attackers have old present/absent transmission patterns of the consumers to be used for training the attacker's deep-learning model to launch PPA.

IV. Preliminaries

A. Deep-Learning

A neural network (NN) composes of three types of layers, input, hidden, and output [15]. Deep-learning is a type of NN which uses two or more hidden layers. The training process of a deep-learning model begins with feeding the input layer of the model with the input data. Next, there are two processes, namely feed-forward and back-propagation, which facilitate mapping the input data through the other layers. These processes are done for a predefined number of epochs, and in each epoch, the weights and biases are updated in the direction of improving the model accuracy [15]. In this work, for training the proposed attacker and defense models, we use convolutional neural network (CNN) and gated recurrent unit neural network (GRU).

B. Convolutional Neural Network (CNN)

CNN is widely used in solving many problems that have time series data [16]. This is because of its ability to capture complex patterns in the input data. The traditional design of a CNN model consists of input, convolution, pooling, fully connected, and output layers. The convolution layer comprises of a group of filters and pooling layer(s) to facilitate extracting the important features from the input. One or more fully connected layers usually come after the pooling layer(s) to make more complex processes on the extracted features used for inference [16].

C. Gated Recurrent Unit Neural Network (GRU)

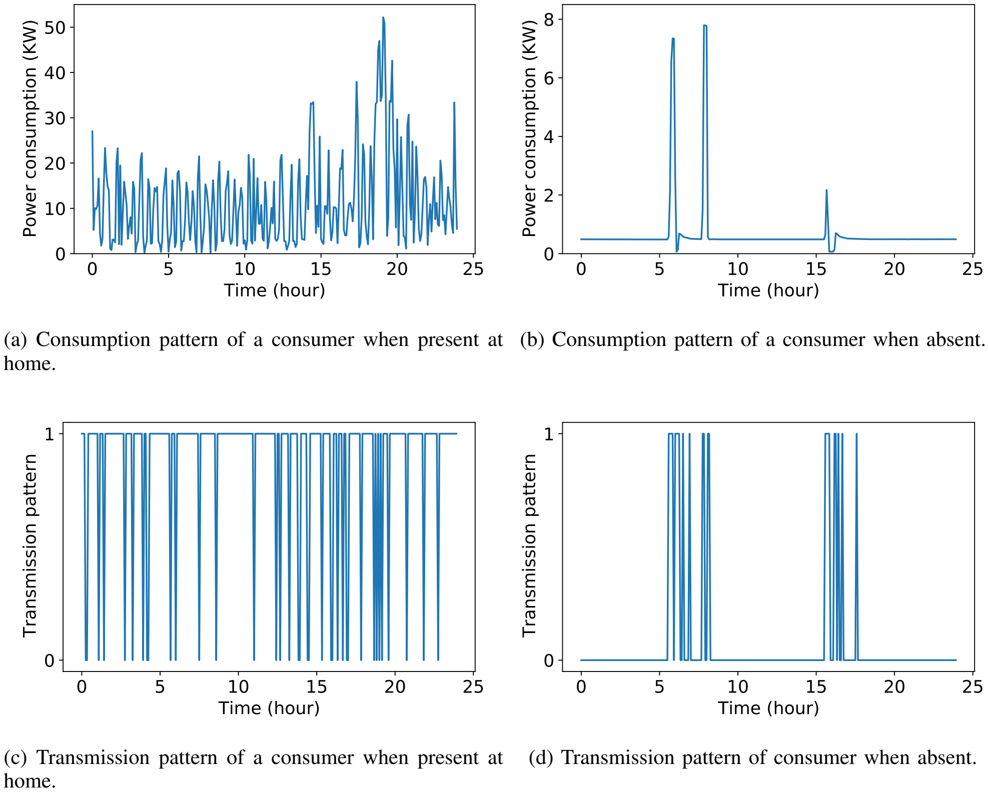
GRU is a kind of recurrent neural networks that utilizes hidden states, often called hidden memory, to memorize long sequences of input data and create connections between internal units [16]. GRU uses the current time and the preceding hidden state information to compute the current hidden state. Therefore, GRU has the potential to find the correlations among the input data, and that is the reason it will be used to train our models.

V. Dataset Preparation

In this paper, a real smart meter dataset X_{SM} is used for training and evaluating our proposed attack and defense models. This dataset was produced by the smart project [17] in 2016 from January 1^{st} to December 14^{th}. It contains real power consumption readings for 114 households in which a power consumption reading is reported every one minute. By processing this data, we build a total number of records of $39,786$, where each record represents a set of power consumption readings of a consumer in a single day (i.e., 1440 readings). We have made another dataset $\hat{X}_{SM_5}$ with a transmission rate of 1/5min from X_{SM}, in which readings are sent every 5 minutes by aggregating the power consumption readings. Then, we used $\hat{X}_{SM_5}$ to create another dataset $\hat{X}_{SM_{CAT}}$ for the CAT approach with the threshold of 10%. This means that an SM sends its reading only when the current consumption is greater or less than the last reported reading by more than 10%.

Due to the unavailability of a power consumption dataset that includes the presence and absence information of the consumers, we propose a labeling approach to be applied to each consumer's record to label it absent or present as follows. First, the day is divided into three intervals, t_1, t_2, and t_3, that correspond to the following time intervals, 8AM-4PM, 4PM-12AM, and 12AM-8AM, respectively. Next, we compute the total consumption of each period t_i, where $i = \{1, 2, 3\}$. Then, we calculate the following formula for each consumer's record.

$$\left|\frac{C_1[d] - C_2[d]}{C_1[d]}\right| + \left|\frac{C_3[d] - C_2[d]}{C_3[d]}\right|,$$



(a) Consumption pattern of a consumer when present at home.

(b) Consumption pattern of a consumer when absent.

(c) Transmission pattern of a consumer when present at home.

(d) Transmission pattern of consumer when absent.

Fig. 2: Consumption and transmission patterns of a consumer when he/she is present at home (a, c) and when absent (b, d), where in (c, d), 1 and 0 refer to transmission and no transmission events, respectively.

where $C_i[d]$ is the total power consumption of the i^{th} period in day d. The intuition behind using this approach is that, as confirmed by [4], the consumer does a lot of activities at period t_2, e.g., cooking, than other periods if he/she present at home. Therefore, the consumption at period t_2 is usually different from the consumption at the other periods if the consumer is presents at home. Hence, if the result of this formula is low, it means that the consumption at period t_2 is close to the consumption at periods t_1 and t_3, and thus the consumer is labeled absent.

After labeling $\hat{X}_{SM_{CAT}}$, we converted the power consumption reading dataset $\hat{X}_{SM_{CAT}}$ into CAT transmission patterns dataset $\hat{X}_{SM_{trans}}$. Each time slot in a record contains zero or one corresponding to no transmission and transmission events, respectively. These events depend on the absolute value of the percentage of change between the current and the last reported consumption, so there is no transmission if the value below the threshold, otherwise, there is a transmission event. Hence, a transmission dataset $\hat{X}_{SM_{trans}}$ is created with each consumer's record of 288 binary features and is labelled *absent* or *present*. The dataset $\hat{X}_{SM_{trans}}$ is then divided into two sets, 80% for training and 20% for testing.

Fig. 2 shows the power consumption and transmission patterns for a randomly selected consumer in our dataset when he/she is present at home and absent. We can observe from the figure that the pattern when the consumer is present at home is significantly different from the pattern when he/she is absent. This fact can be exploited by the attackers to learn the absence of the consumers from their transmission patterns. In next section, we will use our dataset $\hat{X}_{SM_{trans}}$ for training and evaluating both the attacker and defense models.

VI. Proposed scheme

In this section, using $\hat{X}_{SM_{trans}}$, we train an attacker model to launch PPA, and then, we train our proposed defense model to thwart PPA.

A. Attacker Model

In this subsection, we explain how PPA can be launched by training a deep-learning model to infer the absence of a consumer. First, to capture the temporal correlation in the transmission pattern, the attacker uses a CNN-based deep-learning model. Next, using hyperopt tool [18] during the training process of the attacker's model, we adjust the model hyper-parameters to achieve the best performance. Table I presents the attacker's model structure as follows. The input of the attacker's model is the consumer's transmission pattern, and then feeding the input to the convolution and max pooling layers to capture the temporal correlation in the transmission pattern. Four fully connected layers are subsequently used to extract more features and make complex decisions. Finally, a softmax output layer is used to classify whether the consumer is present or absent.

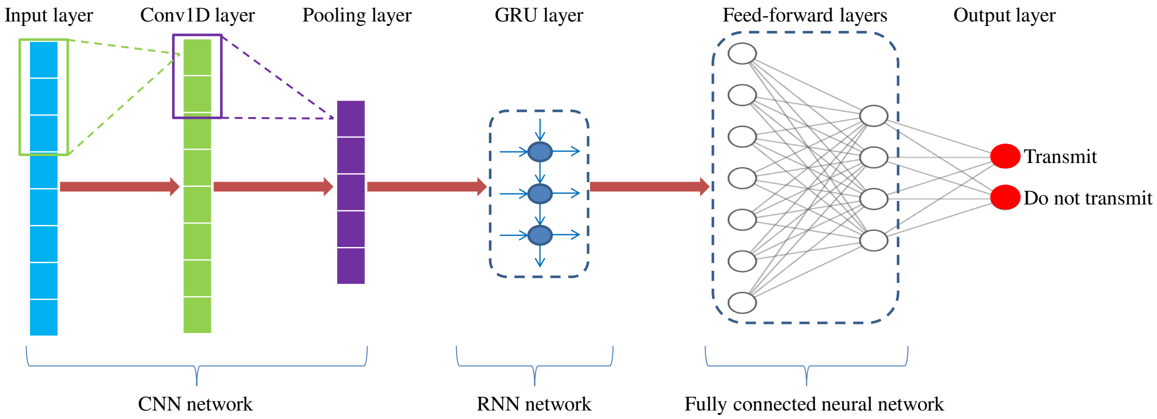


Fig. 3: Illustration of the CNN-RNN-based defense architecture.

TABLE I: The attacker and defense model architectures, where AF stands for activation function, and h_i is the i^{th} fully connected hidden layer.

Layer	No. of units using		AF using	
	Attacker	Defense	Attacker	Defense
Input	288	100	Linear	Linear
Conv1D	128	128	ELu	ReLU
Conv1D	64	–	ReLU	ReLU
Conv1D	64	–	ELu	ELu
Conv1D	32	–	ReLU	–
MaxPooling1D	2	4	–	–
GRU	–	200	–	Tanh
h_1	512	128	ELu	ReLU
h_2	64	32	ReLU	ReLU
h_3	128	–	Sigmoid	–
h_4	64	–	ELu	–
Output	2	2	Softmax	Softmax

Moreover, during the training process of the attacker's model, we used $\ell 2-$regularization to limit over-fitting, *Adam* optimizer, 60 epochs, 128 batch size, 0.001 learning rate, and categorical cross entropy as the loss function. In addition, we used Keras Python library which is installed on a high-performance cluster (HPC) of the Tennessee Tech University.

B. Defense Model

Our objective in this work is to mitigate transmission analysis attacks in case of using the CAT approach in AMI to preserve the consumers' privacy. The proposed defense model decides interactively (i.e., in each time slot) whether the SM should send spoofing transmissions such that the attacker should not identify them and also, it is difficult to differentiate between the transmission pattern generated by our defense model when a consumer is absent and the transmission pattern when the consumer is present at home. In the following, we explain how the proposed defense model generates spoofing transmissions to countermeasure PPA.

1) Dataset preparation for training the defense model

From $\hat{X}_{SM_{trans}}$, we created another dataset $\hat{X}_{SM_{defense}}$ to train our defense model as follows. We first transform the transmission patterns of the consumers when they are present at home into sliding windows with n look-back transmission observations and a look-ahead of size one. Each observation is either one if there is a transmission event or zero if there is no transmission event. Thus, the features of $\hat{X}_{SM_{defense}}$ contain n transmission observations and its label is the next observation in the pattern, i.e., $(n+1)^{th}$ observation.

2) Training the defense model

As can be seen from Fig. 3, our defense model is a combination of a CNN and GRU followed by a fully connected neural network with Softmax output layer. A transmission pattern of the last n time slots is the input of the defense model and the output is the decision of either the SM needs to send a reading (spoofing transmission) or not. *Since our defense model is trained on the patterns when the consumers are present at home, the output pattern will be close/similar to the patterns when they are present at home* so that the attacker cannot distinguish between the two patterns. Different values of n have been examined and it is found that at $n = 100$, the defense model gives good results. The same settings used in training the attacker's model, which are mentioned in Section VI-A, are also used in training the defense model, except that the batch size is 400 and the learning rate is 0.0001. Table I gives the hyper-parameters of the defense model that achieve best performance.

The primary reason for utilizing deep-learning methods rather than shallow learning methods is that deep-learning has a superior learning ability since it can better capture the complex patterns within the data. Furthermore, the combination of CNN and GRU enhances the capability to extract the correlations between the previous transmission pattern and the current transmission decision, which is the key to improve the decision accuracy.

3) Defense methodology

Algorithm 1 explains how our defense methodology works. When the house occupants leave, the defense should be run in each time slot (i.e., interactive) to make a decision of whether the SM should send a spoofing transmission to countermeasure PPA to preserve the consumers' privacy. On the other hand, regardless of the absence of the house occupants, if the current consumption is greater or less than the last reported consumption by more than the threshold, the SM should transmit the updated power consumption reading to the SO. The SM's memory can be used to store the last n transmission decisions to be retrieved by the defense model.

VII. Evaluations

A. Performance Metrics

The following metrics are used to evaluate our models. First, the success rate (SR) is the probability that the attacker successfully detects the absence of the house occupants. The performance of the defense model is better when SR is low. Second, the false alarm rate (FA) is the probability that the attacker believes that there are no occupants at home while they are actually present at home. Third, the efficiency of the CAT approach is measured in terms of the percentage of the readings that are not transmitted comparing to the periodic transmission approach. These metrics are measured as follows.

$$SR = \frac{TP}{TP+FP}, \quad FA = \frac{FP}{TN+FN},$$

$$\text{Efficiency (\%)} = \frac{P_t - R_t}{P_t} \times 100,$$

where, TP, TN, FN, and FP stand for true positive, true negative, false negative, and false positive, respectively. Regarding the efficiency metric, P_t is the number of readings using the periodic transmission approach and R_t is the number of readings sent due to our scheme which uses the CAT approach.

B. Attacker's Success Rate Without Our Scheme

As mentioned in Section III-B, it is assumed that the attacker has old absent/present transmission patterns of the consumers. These data are used for training the attacker's deep-learning model, which is discussed in Section VI-A, to infer whether the house occupants are at home or not by using the consumer's transmission pattern. By using the attacker's model without running the defense model, the attacker's success rate is 89%, as can be seen in Table II. Hence, it can be concluded that the attacker can infer the absence of the occupants with high certainty.

C. Attacker's Success Rate With Our Scheme

In this subsection, the performance of our defense model is evaluated. Using our defense model, the attacker's success rate is considerably reduced from 89% to only 6.12%, as shown in Table II. This significant reduction in the attacker's success rate is due to the ability of our defense model to generate patterns generated when the consumer is not at home that are similar to the patterns when they are present at home, so the attacker cannot classify the generated pattern by our defense model as absent.

Algorithm 1: Procedure of RSPDL Defense Scheme.

```
if The house occupants are at home then
    Set the mode as "at home" // present
else
    Set the mode as "on travel" // absent
end if
$\gamma$ = predefined threshold // threshold (%)
for each time slot do
    $l$ = last reported consumption
    $c$ = current consumption
    min = $l$ - ($l \times \gamma$)
    max = $l$ + ($l \times \gamma$)
    if $c \leq$ min or $c \geq$ max then
        // run the CAT approach if the current
        consumption exceeds a threshold
        Transmit the power consumption to the
        aggregator.
        Store the transmission/decision in the memory.
        $l = c$.
    else
        if The mode is "at home" then
            Do not send a transmission and store
            the decision in the memory.
        else            // run the RSPDL
            Retrieve the last $n$ transmission decisions
            stored in the memory.
            Input these transmissions into a CNN-RNN
            model to decide whether the SM needs
            to send a redundant packet to the aggregator
            or not.
            if the decision is transmit then
                Transmit a redundant power consumption
                packet to the aggregator.
                Store the transmission/decision
                in the memory.
            else
                Do not send a redundant packet
                and store the decision in the memory.
            end if
        end if
    end if
end for
```

In addition, as shown in Table II, efficiency of 37.9% can be achieved using our scheme with privacy preservation comparing to 69.67% without preserving the privacy. This reduction in efficiency is because of sending more transmissions to preserve privacy, and this can be considered the cost of privacy preservation. Note that reducing the number of transmissions leads to better efficiency but with higher attacker's success.

Furthermore, we evaluate the attacker's performance using the receiver operating characteristic (ROC) curve as shown in Fig. 4. At false alarm of 0.05, this value is acceptable to the attacker as confirmed by [4], the attacker can achieve a fairly

TABLE II: Evaluations with and without our defense model for 1/5min transmission rate at 10% threshold.

	Without our scheme	With our scheme
Attacker's success rate	89%	**6.12%**
Efficiency	69.67%	**37.9%**

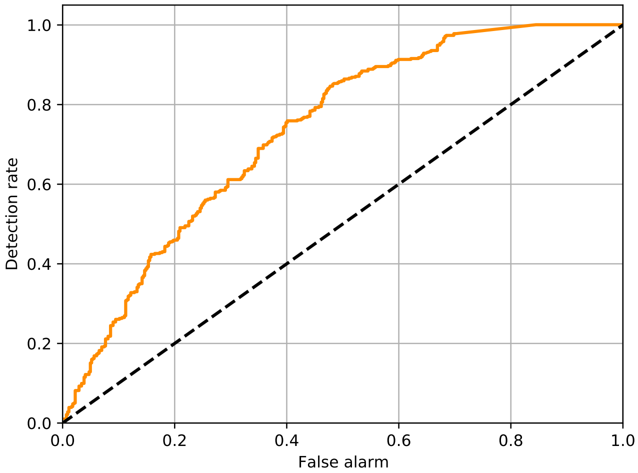


Fig. 4: The ROC curve of the attacker model using 1/5min transmission rate at 10% threshold.

low success rate (around 0.1) comparing to more than 0.6 using ASP proposed in [4]. This means that using ASP [4], the attacker can infer the absence of the house occupants with more confidence, while with using our scheme, the success rate is much lower.

VIII. Conclusion

In this paper, we have proposed a scheme, called "STID", for collecting the power consumption readings efficiently in AMI networks while preserving the consumers' privacy by transmitting spoofing transmissions based on an interactive deep-learning defense model. Our scheme is efficient because there is no need to send the readings if the change in the power consumption is not enough compared to the lasted reported reading. First, we create a dataset that contains CAT transmission patterns using real power consumption readings and a clustering technique. Next, we train a deep-learning-based attacker model to launch PPA, and the results show that the success rate of the attacker is about 90%. Finally, to mitigate the PPA, we train a deep-learning defense model to transmit spoofing transmissions. The evaluations of our envisioned STID scheme demonstrate significant reduction in the attacker's success rate while achieving high efficiency in terms of the number of readings that should be transmitted. Furthermore, our measurements show that our proposed STID can reduce the attacker's success rate to 6.12% and increase the efficiency by about 38% compared to transmitting readings periodically.

Acknowledgement

The authors extend their appreciation to the Deputyship for Research & Innovation, Ministry of Education in Saudi Arabia for funding this research work through the project number 589.

References

[1] M. M. Badr, M. I. Ibrahem, M. Mahmoud, M. M. Fouda, and W. Alasmary, "Detection of False-Reading Attacks in the AMI Net-Metering System," *arXiv preprint arXiv:2012.01983*, 2020.

[2] A. Alsharif, M. Nabil, S. Tonyali, H. Mohammed, M. Mahmoud, and K. Akkaya, "EPIC: Efficient privacy-preserving scheme with EtoE data integrity and authenticity for AMI networks," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3309–3321, 2018.

[3] I. Yilmaz, K. Kapoor, A. Siraj, and M. Abouyoussef, "Privacy protection of grid users data with blockchain and adversarial machine learning," *arXiv preprint arXiv:2101.06308*, 2021.

[4] H. Li, S. Gong, L. Lai, Z. Han, R. C. Qiu, and D. Yang, "Efficient and secure wireless communications for advanced metering infrastructure in smart grids," *IEEE Transactions on Smart Grid*, vol. 3, no. 3, pp. 1540–1551, Sep. 2012.

[5] M. I. Ibrahem, M. M. Badr, M. M. Fouda, M. Mahmoud, W. Alasmary, and Z. M. Fadlullah, "PMBFE: Efficient and Privacy-Preserving Monitoring and Billing Using Functional Encryption for AMI Networks," *Proc. of the International Symposium on Networks, Computers and Communications (ISNCC)*, pp. 1–7, Oct. 2020.

[6] I. Yilmaz and A. Siraj, "A privacy-preserving energy consumption scheme for smart meters with adversarial machine learning," *IEEE Access*, 2021.

[7] A. Alsharif, M. Nabil, A. Sherif, M. Mahmoud, and M. Song, "MDMS: Efficient and privacy-preserving multidimension and multisubset data collection for AMI networks," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10 363–10 374, 2019.

[8] M. I. Ibrahem, M. Nabil, M. M. Fouda, M. M. E. A. Mahmoud, W. Alasmary, and F. Alsolami, "Efficient privacy-preserving electricity theft detection with dynamic billing and load monitoring for AMI networks," *IEEE Internet of Things Journal*, vol. 8, no. 2, pp. 1243–1258, Sep. 2020.

[9] K. Samarakoon, J. Ekanayake, and N. Jenkins, "Reporting available demand response," *IEEE Transactions on Smart Grid*, vol. 4, no. 4, pp. 1842–1851, Dec. 2013.

[10] K. Carrie Armel, A. Gupta, G. Shrimali, and A. Albert, "Is disaggregation the holy grail of energy efficiency? the case of electricity," *Energy Policy*, vol. 52, pp. 213 – 234, Jan. 2013.

[11] S. Werner and J. Lunden, "Event-triggered real-time metering in smart grids," *Proc. of European Signal Processing Conference (EUSIPCO)*, pp. 2701–2705, Sep. 2015.

[12] S. Werner and J. Lundén, "Smart load tracking and reporting for real-time metering in electric power grids," *IEEE Transactions on Smart Grid*, vol. 7, no. 3, pp. 1723–1731, May. 2016.

[13] A. Proano, L. Lazos, and M. Krunz, "Traffic decorrelation techniques for countering a global eavesdropper in WSNs," *IEEE Transactions on Mobile Computing*, vol. 16, no. 3, pp. 857–871, Mar. 2017.

[14] S. Alsemairi and M. Younis, "Adaptive packet-combining to counter traffic analysis in wireless sensor networks," *Proc. of International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 337–342, Aug. 2015.

[15] S. Chaturvedi, R. N. Titre, and N. Sondhiya, "Review of handwritten pattern recognition of digits and special characters using feed forward neural network and Izhikevich neural model," *Proc. of International Conference on Electronic Systems, Signal Processing and Computing Technologies*, pp. 425–428, Jan. 2014.

[16] W. Yin, K. Kann, M. Yu, and H. Schütze, "Comparative study of CNN and RNN for natural language processing," *arXiv preprint arXiv:1702.01923*, 2017.

[17] Kolter, "Residential Energy Disaggregation Dataset (REDD)," Last accessed: 2020. [Online]. Available: http://traces.cs.umass.edu/index.php/Smart/Smart

[18] J. Bergstra, B. Komer, C. Eliasmith, D. Yamins, and D. D. Cox, "Hyperopt: a Python library for model selection and hyperparameter optimization," *Computational Science & Discovery, doi: https://doi.org/10.1088/1749-4699/8/1/014008*, 2015.