Deploying and Evaluating Pufferfish Privacy for Smart Meter Data (Technical Report '15)

Stephan Kessler, Erik Buchmann, and Klemens Böhm
Karlsruhe Institute for Technology (KIT), Am Fasanengarten 5, Karlsruhe, Germany
stephan.kessler@kit.edu, erik.buchmann@kit.edu, klemens.boehm@kit.edu

ABSTRACT

Information hiding ensures privacy by transforming personalized data so that certain sensitive information cannot be inferred any more. One state-of-the-art information-hiding approach is the Pufferfish framework. It lets the users specify their privacy requirements as so-called discriminative pairs of secrets, and it perturbs data so that an adversary does not learn about the probability distribution of such pairs. However, deploying the framework on complex data requires application-specific work. This includes a general definition of the representation of secrets in the data. Another issue is that the tradeoff between Pufferfish privacy and utility of the data is largely unexplored in quantitative terms. In this study, we quantify this tradeoff for smart meter data. Such data contains fine-grained time series of power-consumption data from private households. Disseminating such data in an uncontrolled way puts privacy at risk. We investigate how time series of energy consumption data must be transformed to facilitate specifying secrets that Pufferfish can use.We ensure the generality of our study by looking at different information-extraction approaches, such as re-identification and non-intrusive-appliance-load monitoring, in combination with a comprehensive set of secrets. Additionally, we provide quantitative utility results for a real-world application, the so-called local energy market.

1. INTRODUCTION

Designing a smart grid electricity-supply infrastructure is an important issue. This is because it promises to reduce CO_2 emissions and to guarantee supply at affordable prices. The smart grid initiative requires the installation of smart meters in private households. These devices measure the power consumption in short time intervals, e.g., every 15 minutes. Thus, they produce time series that contain the sum of the energy consumption of all electrical devices active during such a time interval. Various applications require access to the this data. Think of demand-side management or local energy markets [7], an efficient way of allocating renewable energy. They require access to the entire time series. Legal obligations like the EUC 543/2013 even mandate the publication of market data. However, privacy regulations and privacy preferences of individuals are in the way of arbitrary parties accessing such data. Obligations such as the European Directive 95/46/EC only allow the disclosure of data only if it is non-personalized, or if the individuals have consented.

Smart-meter data contains a lot of personal information [24, 3, 15, 6]. This includes information on devices running and on the presence of residents. In consequence, any smart grid service must consider the antagonism between the disclosure of personal data and the privacy of individuals. Which information actually is considered private depends on the individual. Thus, processing time series while protecting privacy requires privacy constraints that one can define individually. Libraries of constraints one can resort to are conceivable as well. The information to be hidden is referred to as *secrets*. Potential secrets go well beyond aggregated values from several households approaches such as [2] have exclusively focused on so far. De-personalization of such data ('anonymization') is not applicable in many cases either: Work on re-identification [6] shows that it is very difficult to remove all relationships to individuals from smart-meter data while preserving utility. Furthermore, use cases such as demand-side management do require data with identifiers.

Example 1: Bob has a smart meter and is willing to accept the disclosure of his data if it does not contain certain information. Suppose that Bob has a flow heater which starts when he begins showering, stops when he finishes and does not consume power otherwise. This heater will be our running example. Bob wants to keep the exact time private when he is showering on weekends and in the morning during weekdays. This defines the secrets. An adversary should not be able to learn whether the flow heater is starting or stopping between 8:00 and 11:00 on a weekday by inspecting the disclosed data. On weekends, the data should be so noisy that inferring the time when the heater is working is unlikely. To this end, one has to know how the time series reflect the heater usage and hide this on a weekday and detect when the heater starts and stops on a weekend. Approaches such as applying differential privacy on smart meter data [2] do not help with this kind of secret. Finally, to preserve utility the data should still contain information that Bob does not explicitly want to hide. □

Individuals might allow the disclosure of their smart-meter data if their privacy preferences are strictly respected. Each individual should have the option to specify such private information. The Pufferfish privacy framework [20] guarantees that certain sensitive information is removed from a data set. Pufferfish supports the definition of intuitively understandable privacy requirements and their semantics. It also covers correlations within the data set, which is sometimes necessary to guarantee privacy while keeping utility. Differential privacy in turn leaves aside such correlations.

Example 2 (*Correlations in the data*): Let $f(A), f(B), f(C)$ be smart-meter time series of Alice, Bob and Carl's household. $f(B)[t]$ is the total power consumption of Bob's household at time slot t. Differential privacy approaches [2, 29] publish the privacy-enhanced sum at each time slot of the households considered, i.e., $f(B)[t] + f(A)[t] + f(C)[t] + \ldots$: If there is not any correlation of the consumptions of Bob, Alice and Carl, an adversary cannot infer the actual consumption of one of them. However, there also are correlations when looking at each time series in isolation: Suppose that Alice, Bob and Carl each have a flow heater (for the shower) and bath lighting. $f(B)^1[t]$ is Bob's flow heater consumption and $f(B)^2[t]$ the one of the bath lighting. $f(B)[t]$ is the sum of all appliances in Bob's household: $f(B)[t] = f(B)^1[t] + f(B)^2[t] + \ldots$. Privacy cannot be guaranteed in the same way as for the sum of $f(B)[t], f(A)[t]$ and $f(C)[t]$: The flow heater and the bath lighting obviously have correlations Differential Privacy does not deal with [19]. □

Pufferfish is an abstract framework that, regarding smart-meter data, (i) requires challenging conceptual work and (ii) has not been evaluated quantitatively. The challenges are to represent private information in smart-meter data, to perturb the aggregated data according to Pufferfish guarantees, to ensure generality and to evaluate utility and coverage of privacy requirements. Regarding (ii), we examine the trade-off between privacy and utility in the scenario in particular.

Representation of Private Information. Each time a specific device runs, this results in a sequence of power-consumption values added to the total consumption. Such sequences corresponding to runs of the same device may vary in the actual values. This is because (a) appliances have a slightly different power consumption each time they run, and (b) the smart meter may measure their consumption together with the ones of other devices. A first challenge not covered by currently existing approaches is to find an abstracted representation of time series flexible enough to cover this uncertainty and specific enough to have a meaning for the secret in question. We call a single value of such an abstracted representation coefficient. This abstraction must have a clear-cut semantics, and the transformation of the time series to this representation must be well-defined. The goal of the abstraction is to have coefficients with a meaning allowing to formulate specific secrets: One should choose transformations whose results correspond to potential secrets.

Example 3 (*Flow heater, abstraction and coefficients*): In Example 1 the coefficients have to allow conclusions regarding the heater. Suppose that a heater consumes $25kW$ when running and $0W$ otherwise. Thus, a difference of the power consumption at point of time t to $t + 1$ of around $25kW$ possibly indicates a starting flow heater. Exactly this can be subject of a privacy requirement. A meaningful abstraction then has a coefficient representing this kind of change. While the start of the flow heater results in two successive consumption values, other devices will create more complex sequences. For example, a washing machine carries out different tasks like heating or spinning. Such information must be hidden if it is relevant for someone's privacy. □

Perturbing Smart-Meter Data. Pufferfish requires to adapt the data that represents a secret. This is not straightforward, because provable privacy guarantees require perturbations that fulfill the Pufferfish requirements [20]. In particular, perturbing an aggregate of several appliances is not obvious, since it requires a decomposition on a conceptual level. Next, we must take into account that different appliances in the decomposed representation may have correlations. Our objective is to deal with such time series individually per appliance.

Generality. It is challenging to find a suitable abstracted representation of the secrets so that the semantics of the representation (a) covers a wide range of privacy requirements for smart meter data and (b) allows Pufferfish to prove compliance with these requirements.

Evaluation. Quantifying the usefulness of the perturbed smart meter data is not obvious: General abstract distance measures for time series do not necessarily quantify utility.

Example 4 (*Abstract Distance measures*): Suppose that a time series is perturbed two times. Further, with the second perturbation, the Euclidean distance of the resulting series to the original one is twice as large as the first one. This does not mean that utility is halved. For example, it may still be possible to identify outliers in time series. □

Next, the evaluation of utility requires meaningful user-defined privacy requirements. Finding realistic requirements is challenging since many individuals are not yet aware of the privacy risks of the smart grid. Thus, an objective source of requirements is needed for a meaningful evaluation.

Contributions. We address all these challenges as follows: Since the kinds of possible secrets are broad, we carefully select different abstracted representations together with adequate transformations for each of them. We illustrate this using the wavelet transformation as example; it covers several kinds of possible secrets. Privacy is guaranteed by the decomposition of the aggregated power signal into several channels on a conceptual level and the application of noise following the ϵ-Pufferfish principle [20] . Before publication, a time series is transformed back to the time-based representation. Thus, the published privacy-enhanced and the original time series have the same format.

In our evaluation, we show that this transformation principle is general enough to cover a wide range of requirements. We arrive at objective privacy requirements by looking at the outcome of various information-extraction methods from literature, i.e., features of smart metering data that others have deemed relevant. In particular, we define secrets covering a re-identification [6] and a non-intrusive-appliance-load monitoring [4] approach. Next, in a local energy market [7], the utility of participants depends on the accuracy of the description of their demand; using perturbed data instead of the real one is expected to curb utility. Here, utility not only is an objective measure, it also has the nice characteristic that it can be quantified as welfare, an established notion from economics. The impact of privacy guarantees on utility is relatively low, while hiding realistic secrets. In numbers, even with severe secrets that require to modify the entire time series, the welfare in that energy market is reduced by 26% only.

Paper outline: We start with related work (Section 2) and then introduce our way of applying Pufferfish (Section 3). We analyze of different transformations (Section 4) and evaluate our approach (Section 5). Section 6 concludes. – There exists an extended version of this article, containing a more detailed description of Pufferfish and the wavelet transformation, proofs of the lemmas and material that complements the evaluation [18].

2. FUNDAMENTALS

Having defined a common notation in Section 2.1, we review well-known privacy-protection approaches in Section 2.2. The Pufferfish Framework is explained in Section 2.3. The wavelet transformation (Section 2.4) is a technique to process and analyze time series, which we use as well. Other related work in turn is discussed in Section 5.

2.1 Notation

In order to support different abstract representations of time series, we have chosen a vector-based representation. Vectors are elements of a vector space. The coefficients of each vector defined on a basis express a finite linear combination of this basis. In other words, the basis defines the meaning of the coefficients. Vectors also allow to change the basis, resulting in other meanings of the coefficients. The standard representation of a time series is a mapping between points of time and the value domain, e.g., consumption values measured. Thus we need to define the time domain $\mathcal{T}$ first and then define a time series as a vector.

Definition 1 (Time domain $\mathcal{T}$): $\mathcal{T}$ is the standard domain of the time series considered. We assume that it is discrete and of finite length, i.e., $\|\mathcal{T}\| \leq \infty$. □

Definition 2 (Time series): A time series is an n-dimensional vector with the basis B, referred to as f_B. To refer to its t-th element, we write: $f_B[t]$. □

In this work, we refer to time series as vectors, using common vector notation. This requires a definition of a standard basis consisting of canonical unit vectors e_i. For a given $\mathcal{T}$, we define the relationship of a time series f to each $t \in \mathcal{T}$: Let $[t_1, ..., t_n]$ be the ordered list of all $t_i \in \mathcal{T}$. Then $f_B[t_i] = f_B^\top \cdot e_i$ is the electricity consumption at time slot t_i. In other words, e_i represents the ith ordered element of $\mathcal{T}$, and $B = \{e_i | i = 1 \ldots n\}$ forms the standard basis.

Definition 3 (Vector space): $\mathcal{V}_B$ is the vector space containing all linear combinations of basis B. □

2.2 Privacy-Protection Approaches

Next to Pufferfish, which serves as the framework for this current work and is described in Section 2.3, there is further related work. Differential Privacy provides provable privacy guarantees for statistical databases [13] and has been applied to smart meter data [2] and time series [29]. Example 2 has illustrated the limitations. Other approaches for time series disclose only aggregated results [5, 31] or build on k-anonymity [1, 25]. In contrast to such approaches, we are not limited to one specific information-extraction goal. Pufferfish features a more general approach, namely hiding user-defined secrets. Additionally, [5, 31, 1, 25] do not give provable guarantees. The approach evaluated here in turn allows for arbitrary queries over the disclosed data.

There exist privacy approaches applicable to time series built on k-anonymity [1, 25], with its known limitations. The intuition is that an individual is indistinguishable amongst $k - 1$ others. Usually k time series are generalized to a common representation. However, k-anonymity based approaches do not allow for individual preferences. Further, such approaches remove identifiers, making the data useless for applications dependent on these.

A perturbation method which handles each individual time series in isolation is to add random noise. However, there exist several methods to de-noise time series and to recover the original values, see [26]. As a counter-measure to denoising techniques, the perturbation scheme in [26] transfers the time series to a Fourier or wavelet representation and then adds noise to coefficients exceeding a threshold. However, a data owner cannot decide what exactly is perturbed. This may result in unnecessarily perturbed information and in sensitive information still present.

Another approach for protecting privacy in smart-meter data is to install batteries and to introduce privacy-aware power routing strategies [23, 17]. However, this requires installation of additional hardware, and privacy requirements may not exceed battery capacities. [28] explicitly considers the privacy-utility tradeoff for smart-meter data, but without formal guarantees.

2.3 The ϵ-Pufferfish Framework

Pufferfish [20] is a generalization of Differential Privacy providing provable privacy guarantees and utility [19]. Pufferfish requires the definition of the following constituents: (a) A set of potential secrets $\mathcal{S}$. $\mathcal{S}$ describes *which* information can be hidden. It is a domain for $\mathcal{S}_{pairs}$. (b) The discriminative pairs of secrets $\mathcal{S}_{pairs}$, describe *how* a piece of information should be hidden. (c) Pufferfish requires data-evolution scenarios $\mathcal{D}$.

Pufferfish privacy means hiding specified secrets $\mathcal{S}$. Examples for the relational data model are: 'Bob has cancer.' or, on another abstraction level, 'The record of individual i is in the data.'. In general, secrets are facts an individual wants to hide. $\mathcal{S}_{pairs}$ is a subset of $\mathcal{S} \times \mathcal{S}$. Pairs of secrets specify *what* an adversary should not be able to distinguish. For example, an individual does not want an adversary to know whether she has cancer or not, so the corresponding pair would be ('Alice has cancer.','$\neg$ Alice has cancer.'). The framework features privacy guarantees for discriminative pairs (s_i, s_j). This is advantageous, as it requires less noise to hide the specific kind of cancer Alice has, compared to hiding whether she has cancer at all. Discriminative pairs have to be mutually exclusive, i.e., at most one is true, but not necessarily exhaustive, i.e., both can be false.

Data-evolution scenarios contain assumptions on how the data has been generated. This is background knowledge of an adversary. It quantifies how likely a fact is. For example, if a data set is from a cancer center, the probability that a patient has cancer is higher than for a normal hospital. Technically speaking, $\mathcal{D}$ is a set of probability distributions over the possible database instances $\mathcal{I}$. Each $d \in \mathcal{D}$ corresponds to the background knowledge of an attacker on how the data has been generated. For example, $P(\mathcal{D}ata = \{x_1, ..., x_n\} | d_p) = p(x_1) \cdot ... \cdot p(x_n)$ if the probabilities of each record in $\mathcal{I}$ are independent. $P(\mathcal{D}ata = \{x_1, ..., x_n\} | d_p)$ is the conditional probability that $\mathcal{D}ata$ is $\{x_1, ..., x_n\}$ under d_p.

A privacy mechanism $\mathcal{M}$ is a method for transferring a data set $\mathcal{D}ata$ into a perturbed and privacy-enhanced representation $\mathcal{M}(\mathcal{D}ata)$. It guarantees the ϵ-Pufferfish privacy criterion if it fulfills the following definition:

Definition 4 (ϵ-Pufferfish Privacy): Given a set of secrets $\mathcal{S}^{\mathcal{P}}$, a set of discriminative pairs $\mathcal{S}^{\mathcal{P}}{}_{pairs}$, data-evolution scenarios $\mathcal{D}$ and a privacy parameter $\epsilon > 0$, a privacy mechanism $\mathcal{M}$ satisfies ϵ-Pufferfish$(\mathcal{S}, \mathcal{S}_{pairs}, \mathcal{D})$-Privacy if, for all outputs of $\mathcal{M}$, all pairs $(s_i, s_j) \in \mathcal{S}_{pairs}$ and all distributions $d \in \mathcal{D}$ the following holds:

$$P(\mathcal{M}(\mathcal{D}ata) = o|s_i, d) \leq e^{\epsilon} \cdot P(\mathcal{M}(\mathcal{D}ata) = o|s_j, d)$$

$$P(\mathcal{M}(\mathcal{D}ata) = o|s_j, d) \leq e^{\epsilon} \cdot P(\mathcal{M}(\mathcal{D}ata) = o|s_i, d)$$

$P(\mathcal{M}(\mathcal{D}ata) = o|s_j, d)$ is the probability that the output of $\mathcal{M}$ is o if s_j holds, and the data distribution is d. □

The intuition is best explained with the following equation that is directly computed from Definition 4:

$$e^{-\epsilon} \leq \frac{P(s_i|\mathcal{M}(\mathcal{D}ata) = o, d)}{P(s_j|\mathcal{M}(\mathcal{D}ata) = o, d)} \Big/ \frac{P(s_i|d)}{P(s_j|d)} \leq e^{\epsilon}$$

If an adversary thinks that s_i is α times as likely as s_j, then, after having access to the privacy enhanced output of $\mathcal{M}$, he may only believe that s_i is at most $e^{\epsilon}\alpha$ times and at least $e^{-\epsilon}\alpha$ as likely as s_j. The framework itself only specifies the privacy guarantees and does not require a specific perturbation method, as long as the guarantees are fulfilled.

2.4 Wavelet Transformation

We use the wavelet-transformed representation as an example, in order to express secrets and to hide them. The following is a concise review, see for instance [27] for a comprehensive introduction. Note that our study is not limited to the wavelet transformation, see Section 4.

Definition 5 (Wavelet): A wavelet $w[t]$ is a finite time series with properties: $\int_{-\infty}^{+\infty} w[t] = 0$ and $\int_{-\infty}^{+\infty} w[t]^2 = 1$. □

Definition 6 (Wavelet Transformation): A wavelet transformation is an orthonormal basis transform to a wavelet basis. Each element of the wavelet basis is a development over time. □

To cover the whole n-dimensional vector space, the wavelet transform results in multiple levels, reflecting different horizontally stretched representations of $w[t]$. Further, the wavelet transformation is invertible. The coefficient at the highest level, the *scaling coefficient*, is not a multiple of the wavelet $w[t]$; it represents the absolute y-position of the time series.

Figure 1 contains a graphical representation of the Haar wavelet used in the paper. Its form indicates, that a time series in Haar Basis always results in coefficients reflecting 'changes' between consecutive points of time. Definition 5 holds since the area under the curve and the one above are of equal size. Wavelet transformation constructs a basis consisting of orthonormal basis vectors of time shifted and stretched wavelets. The Haar wavelet basis contains for instance a vector represented by Figure 1 in the time domain. If we transform f to a basis consisting of (Haar) wavelets f_w, each element $f_w[x]$ represents the change between neighboring values in the time domain. Generally speaking, f_w now represents the pattern of Figure 1. This intuitive explanation leaves aside that a wavelet as is does not cover the entire vector space since it is considered to be 'short'. However, this is necessary to provide invertibility. To do so, a wavelet transformation results in multiple levels. This leads to a 'horizontally stretched' version of the Haar wavelet. The number of levels depends on the dimensionality/length of the time series.

The first level always represents the wavelet as is. The higher the level, the more 'horizontally stretched' the wavelet becomes. In the second level, a representation with Haar basis represents the change between $f[t]$, $f[t+1]$ and $f[t+2]$, $f[t+3]$, etc. The last level is responsible for the absolute level of the time series and does not correspond to any change. From a signal processing perspective, lower levels contain the high frequencies and higher levels lower ones. Further, the wavelet-transformed coefficients always correspond to a fixed number of time-based coefficients. Thus the transformation keeps their time location.

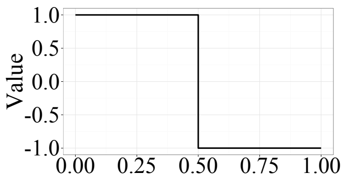

Figure 1: Haar Wavelet

Note that, to ease presentation, we include all the necessary information for the transformation in w. In our example, w contains the Haar wavelet $w[t]$ together with the transformation. An example Haar wavelet transform of the time series on Figure 4 is displayed in Figure 5. A value smaller than zero corresponds to an increasing power consumption. Depending on the position of the increase, the change influences the first or the second level.

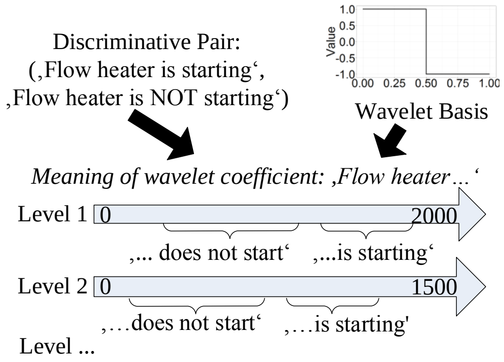

Figure 2: Example: Meaning of wavelet coefficients

Using wavelets requires specifying which elements in f_w are relevant for the individual: Switching on the flow heater (when Bob starts showering) results in a strong sudden increase of the power consumption. In the Haar wavelet domain this leads to high coefficients on lower levels. When the flow heater is switched off, this has an analogous effect on the coefficients. This allows the distinction whether Bob starts/stops to shower or not, cf. Figure 2.

3. PROVABLE PRIVACY FOR SMART METER TIME-SERIES

We now explain our instantiation of the Pufferfish mechanism $\mathcal{M}$ for smart-meter data. $\mathcal{M}(f)$ reconstructs a time series f into one that guarantees ϵ-Pufferfish privacy. We conduct the steps listed in Figure 3. To ease presentation, we assume a single pair of discriminative secrets s_{pair} and a single time series f in what follows. This is not a restriction

```
Input: time series $f$
Input: Set of discriminative pairs $\mathcal{S}_{pairs}$ of secrets $\mathcal{S}$,
       (Inverse) Transformation Mechanism $\mathcal{C}^{trans}_{B'}$,
       $\mathcal{IC}_{B'}$ and basis $B'$
Input: Data evolution scenarios $\mathcal{D}$
Input: Privacy parameter $\epsilon$
Result: Time series with privacy guarantees $f'$
foreach $s_{pair} \in \mathcal{S}_{pairs}$ do
    // Step 1: Transformation;
    $f_{B'} = \mathcal{C}^{trans}_{B'}(f)$;
    // Step 2: Perturbation;
    Determine $\mathcal{N}_\epsilon$ to fulfill $\epsilon$-Pufferfish Privacy based on
    $\mathcal{D}$ and $s_{pair}$;
    Set $p^{coeff}$ according to $s_{pair}$;
    $f'_{B'} = \mathcal{P}(f_{B'}, \mathcal{N}_\epsilon, p^{coeff})$;
    // Step 3: Inverse Transformation;
    $f' = \mathcal{IC}_{B'}(f'_{B'})$
end
return $f'$;
```

Algorithm 1: Pufferfish Privacy Mechanism $\mathcal{M}$ for Smart-Meter Data

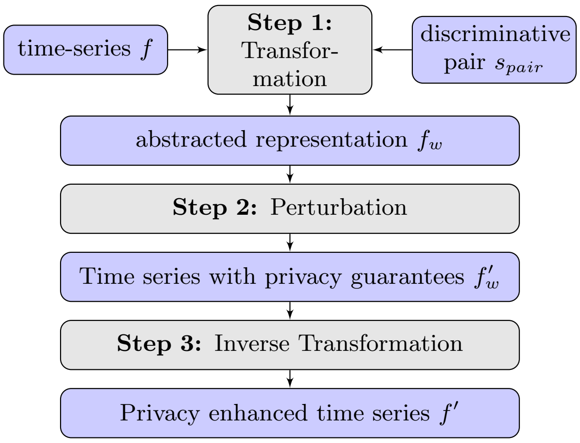
time-series f
Step 1: Transformation
discriminative pair s_{pair}
abstracted representation f_w
Step 2: Perturbation
Time series with privacy guarantees f'_w
Step 3: Inverse Transformation
Privacy enhanced time series f'

Figure 3: Privacy preservation for s_{pair}

since each element of $\mathcal{S}_{pairs}$ is handled in isolation for each time series. Consequently, when speaking of an aggregate, we always mean $f[t]$, the aggregate consumption of all running appliances. For further explanations see Algorithm 1. It contains the pseudo-code including the necessary parameters. We now explain the loop body of Algorithm 1.

Step 1. We transform a time series f to an abstracted representation f_w. Reconsider Example 3. The start of a flow heater requires two consecutive values in the time-based representation. In the Haar transform output in turn, one coefficient is enough to represent this. See Section 3.1.

Step 2. In the transformed representation, secrets determine the perturbation of the abstracted time series according to Pufferfish guarantees. See Section 3.2.

Step 3. We transform the modified time series back to a time based representation f', see Section 3.3.

3.1 Step 1: Transformation

This step transforms a given time series to an abstracted representation where each value carries a specific meaning in relation to secrets (and not necessarily to a point of time). Secrets are geared to specific transformations. Thus we first need to define the transformation mechanisms (Section 3.1.1), before formulating secrets respectively discriminative pairs for smart meter data (Section 3.1.2).

3.1.1 Transformation Mechanism

Representations of time series in an abstracted manner are numerous [11]. The right choice depends on the privacy requirements. Thus, we define requirements on transformation approaches to be applicable with our approach.

Definition 7 (Transformation Mechanism): Let B be the standard basis and B' a different basis of a vector space. A transformation mechanism $\mathcal{C}_{B'}$ is a function of type $\mathcal{V}_B \to \mathcal{V}_{B'}$ that converts a time series from the time-based representation f to an abstracted representation $f_{B'}$ with basis B' and fulfills the following properties:

1. The transformation is invertible, i.e., there exists an inverse of $\mathcal{C}_{B'}$ We refer to it as $\mathcal{IC}_{B'} : \mathcal{V}_{B'} \to \mathcal{V}_B$.
2. $\mathcal{C}_{B'}$ is an endomorphism for the +-operator. Let f, g be time series, then: $\mathcal{C}_{B'}(f + g) = \mathcal{C}_{B'}(f) + \mathcal{C}_{B'}(g)$

□

Suppose that the time series is an aggregate of power consumptions. The endomorphism property simplifies the perturbation: Noise can be added to certain parts of the aggregate as well as to the aggregate, yielding the same result. Section 3.1.2 explains the importance of this property.

The invertibility property implies the following: First, if $f_{B'}$ is invertible, any information of f is present in $f_{B'}$. Thus, any information of f is also included in the abstracted representation. Second, invertibility requires well-defined semantics of every element in $f_{B'}$. Consequently, such clear semantics also hold for secrets dependent on the coefficients, i.e., each coefficient has a specific meaning in relation to a secret. Note that we do not make any restriction on the length of $f_{B'}$ in comparison to f; so the transformation output may also have a higher dimensionality than f.

Haar-Wavelet example transformation. The wavelet transformation as described in Section 2.4 satisfies Definition 7. This transformation for the Haar basis is invertible and an endomorphism for addition. See Lemma 1. Additionally, the wavelet transformation keeps the time location; each value in $f_{B'}[x]$ corresponds to a specific number of entries in $f[t]$. We refer to the wavelet-transformation mechanism with the Haar basis as $\mathcal{C}^{Wave}_h$.

Lemma 1: *The Haar wavelet transformation is invertible and an endomorphism for the +-operator*

Proof: There exists an orthonormal basis for the haar wavelet transformation [12] for any vector with 2^n coefficients. The orthonormal basis vector form a basis transformation matrix H, and the following holds:

$$f \cdot H = f_h$$

This operation is invertible since for each matrix consisting of orthonormal vectors an inverse H' such that $H \cdot H' = I$ exists:

$$f_h \cdot H' = f \cdot H \cdot H' = f \cdot I$$

Additionally, matrix vector multiplication is distributive:

$$f \cdot H = (f^1 + \cdots + f^i) \cdot H = f^1 \cdot H + \cdots + f^i \cdot H$$

Thus, the Haar wavelet transformation is also a +-endomorphism. □

3.1.2 Secrets in Smart-Meter Data

Possible secrets $\mathcal{S}$ an individual may want to hide range from relatively simple ones like *'The dishwasher is running'* to rather complex ones involving several appliances like *'There is cooking activity'*. Other examples are *'There is activity in the kitchen'*, *'The fridge is running'* or *'Someone is watching a certain TV program in the morning'*.

The power-consumption data of a household, usually monitored by a smart meter installed at the main power connection, is the aggregate of all appliances. However, only parts of it typically are relevant for certain secrets. Hence, it is important to be able to examine parts of the aggregate in isolation. Looking at the smart meter time series as a signal, it is the aggregate of several channels. For example, the consumption of the television is one channel $f^1[t]$, the dishwasher is another one, $f^2[t]$.

Definition 8 (Signals and channels): A signal is the complete power consumption measured at the smart meter of the household and is represented as a vector $f[t]$. A channel is a part of the signal, referred to with a superscript, e.g., $f^i[t]$. We see a signal as the sum of n channels: $f[t] = f^1[t] + \cdots + f^n[t]$ □

Even on channels only containing the consumption of individual devices, a sequence of consumption values is still required in many cases to gain interesting information. From non-intrusive appliance load monitoring (NIALM) approaches [15, 22, 21, 14, 4] it is well-known that a sequence of time-value pairs identifies appliances and their state, and appliances tend to be detectable in f.

The connection between values of a time series (even if it is an abstraction) and intuitive descriptions of possible secrets is not obvious. Thus, we define the following.

Definition 9 (Description of a Secret): A description of a secret is a triple

$$s = (s^{Base}, s^{Trans}, s^{Coeff})$$

where s^{Base} is the basis for a transformation mechanism s^{Trans}. s^{Coeff} is the formal description of the coefficients in the abstracted representation $f_{s^{Base}}$ that make s true. We write $f_w[t] \in s^{Coeff}$ if an element of the transformed time series makes the secret true. □

We do not require a specific language to describe the coefficients. However, the description has to be non-ambiguous.

A description of a secret reflects what should be hidden, but not how. It rather is necessary to have discriminative pairs of secrets. Thus, Pufferfish requires a description of discriminative pairs of secrets on smart-meter time series.

Definition 10 (Description of a Discriminative Pair of Secrets): A description of a discriminative pair of secrets s_{pair} is a pair of descriptions of secrets $s_{pair} = (s_1, s_2)$, so that the following holds:

- The base as well as the transformation method are the same ($s_1^{Base} = s_2^{Base}$ and $s_1^{Trans} = s_2^{Trans}$).
- The secrets are mutually exclusive but do not need to be exhaustive, i.e., there may exist values in the range of a coefficient that neither make s_1 nor s_2 true.
- The coefficients in question for s_1 and for s_2 are non-overlapping: $s_1^{Coeff} \cap s_2^{Coeff} = \emptyset$.

□

Typically, only parts of the entire signals are relevant for secrets and discriminative pairs.

Definition 11 (Relevant Channel): For a given signal f consisting of $i \in [1 \dots n]$ channels and for a discriminative pair $s_{pair} = (s_1, s_2)$, we call the channel that contains the information whether s_1 or s_2 is true the relevant channel r. We refer to the corresponding time series as f^r. The decomposition partitions the signal. Formally:

$$f[t] = f^1[t] + \cdots + f^r[t] + \cdots + f^n[t]$$

□

There typically are correlations between channels. They depend on the actual discriminative pair and the assumptions contained in $\mathcal{D}$ regarding an adversary. In Example 2, the lighting f^2 is correlated with the heater f^1. But the lighting consumption is not part of the relevant channel, since it is not directly related to the showering activity.

Correlations result in different data-evolution scenarios and require a different distribution of the noise applied. The specifics are part of the Pufferfish Framework [20].The following example illustrates the description of the secrets in smart-meter time series.

Example 5 (*Instantiations of Secrets for the Heater*): Bob wants to hide whether secret s_1 'The heater is starting/stopping' or secret s_2 'The heater is not starting/stopping' is true. The wavelet transform with the Haar basis reflects 'switch on' respectively 'switch off' events and is suitable for the discriminative pair $s_{pair} = (s_1, s_2)$. Let h be the Haar wavelet basis, then $s_1^{Trans} = s_2^{Trans} = \mathcal{C}_h^{Wave}$. For the sake of simplicity, we assume that the heater power-consumption function is of rectangular shape over time, as illustrated in Figure 4 (generated with the model of [30]). Figure 5 contains $\mathcal{C}_h^{Wave}(f)$ of the time series illustrated in Figure 4: The x-axis in Figure 5 shows the time location and the y-axis the 'intensity' of the Haar basis. Coefficients in Level 1 and 2 reflect the starting and stopping of the heater, as explained in Section 2.4. To include small inaccuracies, we define s_1^{Coeff} to cover coefficients of Level 1 if their value is in $[13, 17]$ or $[-17, -13]$ and Level 2 if their value is in $[18, 22]$ or $[-22, -18]$. Consequently s_2^{Coeff} contains all values of coefficients on Level 1 except for $[13, 17]$ and $[-17, -13]$ and Level 2 except for $[18, 22]$ and $[-22, -18]$. s_1 and s_2 qualify as a discriminative pair s_{pair} since $s_1^{Trans} = s_2^{Trans}$ and $s_1^{Coeff} \bigcap s_2^{Coeff} = \emptyset$. In this example, the channel relevant for s_{pair} only contains the heater consumption. □

For different transformations or for different bases the determination of coefficients works in the same way, as long as the proposed specification of coefficients holds. Using a different transformation or basis allows to cover other requirements, see Section 4.

3.2 Step 2: Perturbation

This section explains how we have ensured Pufferfish privacy in time series of smart meter data. One common method explicitly illustrated in the following is to apply additive Laplace noise to aggregates [20]. If different channels are correlated, the noise should follow other distributions, see [20]. However, this does not affect the following description. As explained in Section 3.1.2, a smart meter signal is an aggregate of different appliances, but noise is only required for some channels. Identifying the channels and the noise distribution applicable is not obvious.

3.2.1 Perturbation Mechanism for Time Series

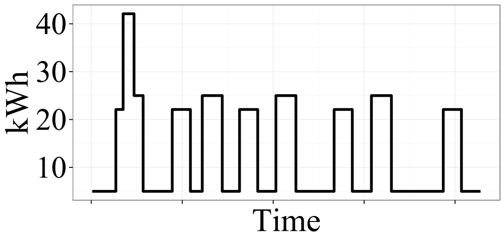

Figure 4: Example consumption time series of a starting/stopping flow heater

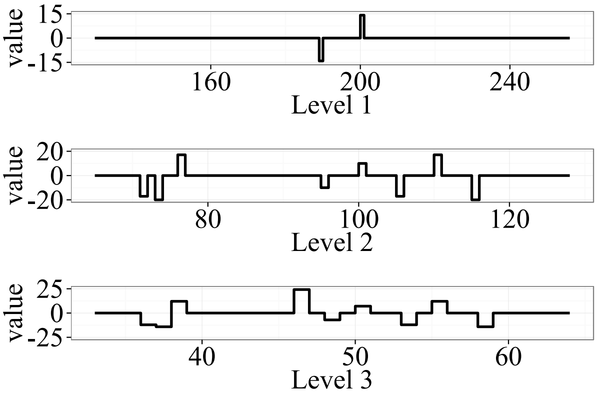

Figure 5: Haar Wavelet decomposition of time series of a starting/stopping heater (only three levels)

We explain our approach for perturbing a time series of smart meter data in the transformed representation. The perturbation naturally must have a noise distribution. We refer to the transformed version with mechanism s^{Trans} and basis s^{Base}, where w consists of s^{Trans} and s^{Base}, as f_w. We refer to the resulting perturbed time series as f'_w.

Additionally to the noise distribution, the perturbation also requires the selection of the coefficients to be noised. This leads to the following definition.

Definition 12 (Perturbation Mechanism for a Discriminative Pair): A perturbation mechanism $\mathcal{P}$ is a function that takes a time series f_w in abstracted representation, the noise $\mathcal{N}_\epsilon$ to be applied dependent on the privacy parameter ϵ and a formal definition of the coefficients to be perturbed p^{coeff}. It returns the privacy-enhanced time series in the transformed representation, referred to as f'_w.

$$f'_w = \mathcal{P}(f_w, \mathcal{N}_\epsilon, p^{coeff})$$

□

3.2.2 Noised elements

p^{coeff} specifies the elements of f'_w to be perturbed. Similarly to the definition of secret descriptions, we leave aside the language for selecting these coefficients. Examples for p^{coeff} are as follows:

- **All:** This is the most simple strategy. Additive noise is applied to all coefficients.
- **Trigger dependent:** Since coefficients in a certain range have a defined meaning, they are perturbed. This is similar to [26]. However, the ranges and the noise have a well-defined meaning (c.f. Figure 2), guaranteeing a certain level of privacy. Note that it is now possible to define the noise relative to $f_w[x]$.
- **Time dependent:** The user specifies coefficients to be perturbed (e.g., from t_1 to t_2 etc.), independent of the value. However, this only works if the transformation mechanism keeps the time location.
- **Trigger and time dependent:** This combines both possibilities just mentioned.

3.2.3 Noise Distribution

$\mathcal{P}$ used with noise according to Pufferfish and to the discriminative pair $s_{pair} = (s_1, s_2)$ guarantees privacy.

Lemma 2: *Let f be a time series of smart meter data, $s_{pair} = (s_1, s_2)$ the information an individual wants to hide, $\mathcal{C}_{sBase}$ a transformation mechanism suitable for s_{pair} and $\mathcal{P}$ a perturbation mechanism. There exists a distribution of noise $\mathcal{N}_\epsilon$ with $\mathcal{P}$ for $\mathcal{C}^f_{sBase}$ that satisfies the ϵ-Pufferfish Privacy Definition.*

Proof: Secrets (Definition 9) as well as discriminative pairs (Definition 10) are defined according to the Pufferfish framework. Assume that data evolution scenario $\mathcal{D}$ defines the distribution of values on each channel of the whole signal, including those on the relevant channel for s_{pair}. Since the transformation mechanism $\mathcal{C}_{sBase}$ is an endomorphism for the $+$-operator, the distribution $\mathcal{D}$ also holds for the abstracted representation. If we apply noise $\mathcal{N}_\epsilon$ for $s_{pair} = (s_1, s_2)$ so that the following holds, ϵ-Pufferfish privacy is guaranteed.

$$P(\mathcal{M}(\mathcal{D}ata) = o|s_1, d) \leq e^\epsilon \cdot P(\mathcal{M}(\mathcal{D}ata) = o|s_2, d)$$

$$P(\mathcal{M}(\mathcal{D}ata) = o|s_2, d) \leq e^\epsilon \cdot P(\mathcal{M}(\mathcal{D}ata) = o|s_1, d)$$

According to [20], a suitable distribution of noise can be found for every $\mathcal{D}$ dependent on ϵ. □

The following example illustrates how to choose noise for the starting flow heater appropriately.

Example 6 (*Hiding the start of the heater*): Bob wants to hide the pair $s_{pair} = (s_1, s_2)$ from Example 5. To do so, we carry out the proposed wavelet transformation $\mathcal{C}^{Wave}_w$ with the Haar basis w. Let f^r be the relevant channel for s_{pair}. To ease presentation, suppose that the channels are statistically independent. The coefficients in question for s_1 and s_2 correspond to non-overlapping intervals by definition. For instance, let $f_w[x]$ be a value of Level 1 of the wavelet-transformed representation. If $f^r_w[x] \in [y-k, y+k]$, s_1 is true for $y = 15$ with an imprecision interval of $k = 2$, otherwise s_2. For Level 2 s_1 is true for $y = 20$ and $k = 2$. In this case, we want to prevent an adversary from learning the value of $f^r_w[x]$ by accessing the privacy-enhanced signal $f'_w[x]$. [20] shows that adding noise drawn from the Laplace$(4k/\epsilon)$ distribution with density function $\frac{\epsilon}{8k}e^{-\epsilon|x|/4k}$ guarantees ϵ-Pufferfish privacy for the aggregate as follows: An adversary cannot distinguish whether the value of a single channel is between $y-k$ and $y+k$ or one of the neighboring intervals $[y+k, y+3k)$ or $[y-3k, y-k)$. Let X be a random variable drawn from the above distribution and x be the coefficient to hide. We then generate the privacy-enhanced aggregate $f'_w[x]$ as follows:

$$f'_w[x] = f^r_w[x] + f^i_w[x] + \cdots + X$$

Note that adding noise does not require the disaggregation of the signal into several channels, i.e., $f'_w[x] = f_w[x] + X$. Adding noise already ensures Pufferfish privacy.

Since wavelet coefficients are time-located, it is possible to add noise for weekdays between 8:00 and 10:00, cf. Example 1. On the weekends, we add noise during the whole day on Levels 1 and 2. □

3.3 Step 3: Inverse Transformation

The last step transforms the abstracted and perturbed representation f'_w back to the time-based one f'. This is possible, since Definition 7 requires invertibility.

4. TRANSFORMATIONS

After having applied ϵ-Pufferfish Privacy on smart-meter data, there still are issues worth to be discussed. First, we have illustrated the hiding of switch-on/off events of a flow heater with the help of the Haar wavelet transformation. However, there are privacy requirements with a different structure which this transform cannot cover. It can hide certain other requirements, as discussed in Section 4.1. Second, as other secrets may require different transformations, we discuss alternatives to the Haar-wavelet transformation in Section 4.2.

4.1 Applications of Wavelet Transformation

Non-intrusive appliance-load monitoring [15] is a collective term for a number of methods. They try to extract information on devices by monitoring the aggregated power consumption of several devices. Next to [15] there exist other recent approaches [22, 21, 14, 35, 3]. The switch on and off events that can be monitored at the power supply are important to detect the appliances. Running appliances usually correspond to specific activities and thus are likely to be considered as private. Thus, it is promising to hide exactly these events in order to protect the privacy of individuals.

The representation with the Haar basis describes the switch-on/off events well. However, there are two limitations: First, the Haar transformation works only for time series of length 2^n since the wavelet has length 2. Second, it is not trivial to find another basis that describes other patterns. In order to cover other secrets, modifications of the wavelet transformation or completely different transformations may be necessary, as described in the following section.

4.2 Transformation Mechanisms

If a transformation fulfills Definition 7, we can use it to hide discriminative pairs of secrets. It is promising to take the transformation an adversary will use to extract information on the discriminative pair into account. For instance, one may take a NIALM approach [22, 21, 14, 35, 3, 15] and deploy a transformation used there. However, not every secret can be represented in the wavelet transformed representation proposed. Thus, in the following we introduce transformations that could be used instead of the one presented so far. These are the Discrete Fourier Transformation, other wavelet transformations, codebooks and multiresolution analysis.

4.2.1 Decomposed Wavelet Transformation:

The Haar wavelet transformation is capable of transforming a time series if its length is a multiple of 2^n. In general this is not the case, but we can decompose the signal: The decomposed wavelet transformation splits the original signal into different disjoint subsequences and applies the wavelet transformation on each one. This allows independent modifications of different periods of the signal. A popular decomposition is the Ancient Egyptian Decomposition [9].

Lemma 3: *The Decomposed wavelet transformation fulfills Definition 7.*

Proof: Lemma 1 states that a wavelet transformation fulfills the necessary requirements. The decomposed transformation processes distinct parts of the time series and thus it also is invertible and an endomorphism. □

4.2.2 Wavelet-Packet Transformation

The wavelet-packet transform is another wavelet transformation. In contrast to the transformation already proposed, it does not require a specified basis such as the Haar basis. In particular, with the help of a time series representing the pattern of a secret the packet transform is able to compute a suitable basis. The resulting basis is matched to the given time series [10]. The advantage of the packet transform is that it can be used to flexibly create wavelet bases that match patterns well. Such a pre-computed basis is used to transform the signal respectively the channels following the standard wavelet transformation. While the wavelet-packet transformation provides further flexibility, we do not use it in our evaluation in Section 5. This is because other transformations suffice to deal with the secrets featured there.

Lemma 4: *The wavelet-packet transformation fulfills Definition 7.*

Proof: The wavelet packet transformation chooses a custom base for the transformation, as a composition of orthonormal bases. Thus it is invertible. Since the transformation applies the same basis to all the channels the addition of the coefficients is well-defined and thus it also is an endomorphism for the + operator. □

4.2.3 Discrete Fourier Transformation

Oscillations in the power consumption are periodically repeating power demands, e.g., appliances running at fixed times. Oscillations also are a characteristic of the state of appliances, e.g., the frequency of power peaks of a television corresponds to the TV program. The discrete Fourier transformation [27] converts a sequence of samples (this is the time series) to a frequency-decomposed representation of the oscillations described. Thus, this transformation allows to hide periodical events.

Lemma 5: *The DFT fulfills Definition 7.*

Proof: Each coefficient in the Fourier-transformed representation corresponds to certain well defined frequencies. Thus, there exists an inverse transformation [33]. Further, the value of each coefficient is the amplitude of a certain frequency. A sum in the time domain of two time series equals to the sum of all frequency amplitudes. The DFT also is an endomorphism for +. □

4.2.4 Codebooks and Multiresolution Analysis

Individuals might have a certain pattern in mind that they want to hide and then use a multiresolution-codebook representation such as [32] to search for this pattern. In a nutshell, a codebook is a map from keys to patterns (sequences of power-consumption values). The abstracted time series is represented by a sequence of these keys, and each value corresponds to the pattern described by codewords in the codebook. In general, there may be a small difference between the codewords and the actual patterns. Usually, these differences are neglected [32], leading to an inaccurate inverse. Invertibility requires recording these differences. Patterns can also be created by compression algorithms [34, 8] such as LZW that extract similar sequences. Whether such transformations fulfill the requirements of Definition 7 depends on the actual algorithm. A codebook is invertible since it is a unique map. It also is an endomorphism for + if the addition of two keys results in a key representing the addition of the patterns in the time domain.

5. EVALUATION

Our evaluation has two goals, generality and utility: First, an individual should be able to hide arbitrary information. Second, the disclosed data should still be useful while guaranteeing privacy to the extent specified.

Regarding the first issue, to evaluate objectively whether our approach is general enough to cover a broad range of privacy requirements we need a reliable source of such requirements. To our knowledge, such a source for smart meter data does not exist. However, there exist recent approaches extracting various kinds of information on individuals from smart meter data. The information these approaches try to extract can be perceived as information that is worth to be protected, i.e., as privacy requirements. We show that it is possible to define discriminative pairs of secrets suitable for these requirements. The approaches explicitly considered in what follows are a non-intrusive appliance-load monitoring approach (NIALM, Section 5.1) and a re-identification approach (Section 5.2). All in all, we have identified over thirteen categories of secrets. We will show that guaranteeing Pufferfish privacy makes information extraction with those methods much more difficult.

We now preview the second issue of quantifying utility. Abstract time-series-distance measures do not allow for meaningful conclusions regarding the utility of a modified time series for applications. See Example 4. To ensure realistic conditions, we evaluate the utility of a noised, privacy-enhanced data set by means of a local electricity market (Section 5.3).

The approach presented hides user-defined preferences in a time series of smart-meter data. A comparison of our approach with another one regarding utility would only be conclusive if the reference point offered the same extent of privacy; but we are not aware of any such approach.

5.1 Generality: The INDiC NIALM Approach

As a first step of evaluating generality, we assume that individuals want to hide whether a specific appliance is running or not. NIALM approaches allow the extraction of running appliances from the aggregated smart meter signal. While the different NIALM methods are numerous, we choose INDiC [4], a refinement of one of the first methods [15]. Compared to other approaches, it is simple but detects appliances accurately. INDiC assumes that each appliance has a number of states with different extents of power consumption, and an appliance can only be in one state at a time. In this case, disaggregation is a combinatorial optimization problem, namely finding the optimal combination of appliances in different states while minimizing the error.

outlet/appliance	**State 1**	**State 2**	**State 3**
dishwasher	$0W$	$260W$	$1195W$
kitchen	$5W$	$727W$	
kitchen2	$1W$	$204W$	$1036W$
light	$9W$	$113W$	$156W$
microwave	$9W$	$822W$	$1740W$
refrigerator	$7W$	$214W$	$423W$
stove	$0W$	$373W$	

Table 1: States of appliances

Evaluating how well secrets hinder information extraction with INDiC requires a ground truth. It contains whether INDiC is successful when extracting information on running devices. Thus, the creation of the ground truth requires the smart meter signal as well as individual channels of devices to compute success rates. We use the REDD dataset [22], which contains the total power consumption of different households consisting of the two 'main' power signals (smart meter) and a number of isolated channels (electricity outlets) monitored in parallel. The disaggregation together with the subsequent evaluation consists of the following steps:

1. The data set (including both main and appliance channels) is divided into a training and a test set.
2. For each appliance channel available, INDiC determines possible different states by clustering the power-consumption values of the training set.
3. Based on the states identified, the main channels in the test-data set are disaggregated.
4. To evaluate the success of the disaggregation, the results computed are compared to the actual appliance-usage data available from the other channels.

5.1.1 *Application of the Pufferfish Framework*

For the definition of secrets descriptions, we require knowledge of devices: Table 1 shows the results of the training. As a result of the training, INDiC comes up with different states of each appliance by finding frequent power-consumption levels. Each level corresponds to a state, and the number of states may vary contingent on the appliance. The states with the corresponding power level are the external knowledge of an adversary trying to gather information by inspecting the aggregated power -onsumption time series f. INDiC determines running appliances by accounting the total power consumption to states.

W.l.o.g., we assume that the household wants to hide if the light is in State 2 or State 3. Choosing another pair only requires to use other power-consumption levels in the secret. Thus, the description of the secrets is $s_1 =$ 'Light is in State 2' and $s_2 =$ 'Light is in State 3'. INDiC works without modifying the representation of the time series. Hence, we modify the time series as is: $s_1^{Trans} = s_2^{Trans} = id$, and the base is $s_1^{Base} = s_2^{Base} = \mathcal{T}$. According to Table 1, light is in State 2 if $113W$ is not accounted to another appliance and in State 3 if $156W$ is not accounted elsewhere. s_1^{Coeff} contains coefficients that result in $113W$, and s_2^{Coeff} contains coefficients that result in $156W$ unaccounted power. Then the discriminative pair is $s_{pair} = (s_1, s_2)$. INDiC assumes that all appliances have the same probability to be in a specific state, i.e., we can assume that $\mathcal{D}$ is evenly distributed when adding noise. Since the se-

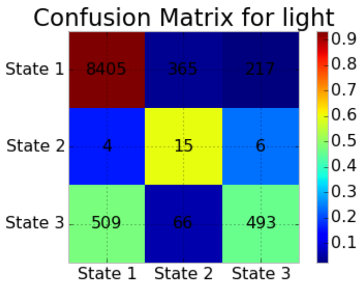

Figure 6: Confusion Matrix for the INDiC approach (without noise)

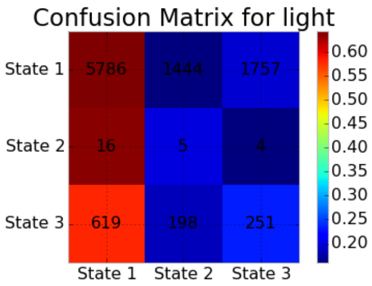

Figure 7: Confusion Matrix for the INDiC approach (with noise)

	State 1	**State 2**	**State 3**
State 1	*0.94*	0.04	0.02
State 2	0.16	*0.60*	0.24
State 3	0.48	0.06	*0.46*

Table 2: Tabular representation of INDiC on data without noise, Predicted states *vs.* True states

	State 1	**State 2**	**State 3**
State 1	*0.64*	0.16	0.20
State 2	0.64	*0.20*	0.16
State 3	0.58	0.19	*0.23*

Table 3: Tabular representation INDiC on data with noise, Predicted states *vs.* True states

crets considered do not specify a time span, we set p^{coeff} to f. To sum up, an adversary should be unable to distinguish whether the unaccounted power is around $113W$ or $156W$. According to Section 3.2.1, we choose Laplace$(4 \times \frac{156-113}{2}/\epsilon)$ noise to perturb the interval between both values. Further, we assume that the household wants to achieve ϵ-Pufferfish privacy with $\epsilon = 0.1$.

5.1.2 Results

In order to quantify the error due the noise we conducted an INDiC disaggregation on the test-data set with and without noise applied. We determine the loss of accuracy as well as the change in uncertainty whether light is in State 2 or State 3. The result is that INDiC guesses the state right for most points of time (Table 2). The rows represent the predicted state of the appliance and the columns the actual state determined as ground truth. Thus, the element at $m \times n$ represents the relative frequency that the m-th state was detected while the state has actually been n. After applying noise, the results get worse (Table 3): Since s_{pair} should hide the distinction between State 2 and 3, we are interested in results covering the probabilities of both. An adversary obviously has difficulties distinguishing which state is true: Guessing the right state is only 4% more likely than guessing the other one (see Table 3). The accuracy drops by 40% regarding State 2 and 23% regarding State 3. This is a massive drop because of our assumption that each state is equally possible. The so-called confusion matrix summarizes the evaluation and provides further insight into the results. It displays the relationship between the states guessed and the actual ones. The rows represent the predicted state of the appliance, and the columns the actual state determined as ground truth. Thus, the element at $m \times n$ represents, how often the m-th state was detected while the state actually was n. Figure 5.1.2 shows the matrix without applying Pufferfish and Figure 5.1.2 with privacy protection.

5.1.3 Limitations

The definition of arbitrary secrets covering other appliances is similar to the case in Section 5.1.1. However, adding Laplace noise with a higher deviation may lead to negative power-consumption values. This obviously is not valid. Replacing negative values with valid ones, e.g., zero, changes the distribution of the noise and thus does not qualify as Pufferfish privacy. One may not be able to guarantee privacy when large differences between states shall be hidden. However, this is not specific to Pufferfish or to this current study. It rather is a general problem of information-hiding approaches: Perturbing information that is a significant part of an aggregated value requires noise with a large variance.

5.2 Generality: Re-Identification

Re-Identification means linking personal data which does not contain any direct identifiers (name, address, etc.) to individuals. Features of the consumption help to re-identify time series of power-consumption values [6]. To illustrate, we focus on the following four features: sum, maximum and minimum of the power consumption for a time interval and average bedtime hour, i.e., the first point of time in the evening when the consumption decreases significantly. Note that we also can hide all other features listed in [6]. Table 4 lists the necessary transformations and the relevant coefficients. Those four features have the same structure as almost half of the features in Table 4.

We now review how re-identification works:

1. The adversary has feature values of households as external knowledge, e.g., a certain household usually goes to bed at $11pm$.
2. For each time series in question, the values for these features are computed. The adversary compares the results with the external knowledge.
3. Assuming that households tend to have repeating behavior over time, features computed for a household for different time periods tend to have similar values. The system computes a score based on the difference between feature values that are part of the external

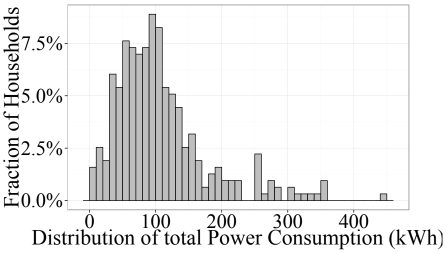

Figure 8: Distribution of the total power consumption

Features	Transformation	Coefficients concerned
Sum	Haar-Wavelet	Scaling Coefficient
Maximum	Haar-Wavelet	Scaling Coefficient
Minimum	Haar-Wavelet	Scaling Coefficient
Evening Sum	Decomposed Wavelet	Relevant Scaling Coeff.
Morning Sum	Decomposed Wavelet	Relevant Scaling Coeff.
0.9 Quantile	Fourier	All
Standard Deviation	Fourier	All
Frequency of mode	Fourier	Significant Frequencies
Fraction of Weekend Consumption	Fourier	Frequencies reflecting fraction
Wakeup time	Haar-Wavelet	Level 1/2
Bedtime	Haar-Wavelet	Level 1/2

Table 4: Feasible Transformation for re-identification features

knowledge and the values of the household in question. The smaller the score, the more likely the household is the sought one.

4. A household is deemed re-identified if its time series receives the n-th lowest score or lower. n is an external parameter and allows to overcome imprecision.

An earlier result is that up to 82.8% of the households can be re-identified [6] in an unmodified data set. To hinder re-identification, certain distinctive features need to be hidden. For the four secrets explicitly considered here, the wavelet transform with a Haar basis is suitable: The scaling coefficient (see Section 2.4) represents the sum and also influences the maximum and minimum, see Section 5.2.1. Levels 1 and 2 reflect the first significant decrease for the bedtime hour, like the heater starting or stopping.

5.2.1 *Hiding Sum, Maximum and Minimum*

Next, we say how the sum, the maximum and the minimum can be hidden. To do so, we take a closer look at re-identification. The total power consumption of a time period is the sum of all channels $i \in [1 \ldots n]$:

$$\sum_{\forall t \in \mathcal{T}} f[t] = \sum_{\forall t \in \mathcal{T}} f^1[t] + \cdots + \sum_{\forall t \in \mathcal{T}} f^n[t]$$

An adversary with external knowledge on the power consumption trying to re-identify a record has to take inaccuracies into account, i.e., he typically does not know the total consumption for sure, only within a certain range. Thus, we partition the channels into a known one, such as the relevant channel r, and the ones not known. The channels not known are responsible for the difference between the known channels and the total consumption at each point of time.

$$\sum_{\forall t \in \mathcal{T}} f[t] = \sum_{\forall t \in \mathcal{T}} f^1[t] + \cdots + \sum_{\forall t \in \mathcal{T}} f^r[t] + \cdots + \sum_{\forall t \in \mathcal{T}} f^n[t]$$

Based on the sum $\sum_{\forall t \in \mathcal{T}} f[t]$ the adversary has to decide whether the known channel is consistent with his knowledge. Adding Laplace noise in line with ϵ-Pufferfish privacy leads to uncertainty regarding $\sum_{\forall t \in \mathcal{T}} f^r[t]$. Re-identification is successful if an adversary is able to single out the true individual record. In particular, this is relatively easy if the feature values of individuals are spread over a wide range and are rather unique. Thus, individual privacy requirements depend on assumptions regarding other individuals in the data set. Describing a suitable secret is deciding which interval is sufficient to hide $\sum_{\forall t \in \mathcal{T}} f^r[t]$ amongst other channels. We use the following notation:

$s_k =$ 'Known power consumption is in interval [y-k, y+k]'

The discriminative pairs can be of the form $s_{pair} = (s_k, s_{3k})$. One way to determine k is to look at the distribution of a known data set. Figure 8 indicates that $k = 5kWh$ is sufficient to hide a single household amongst more than 10 others for a large number of households. These considerations also hold for the features 'Minimum' and 'Maximum'.

Applying noise to the scaling coefficient Applying noise to the scaling coefficient is special, compared to other coefficients. In particular, the scaling coefficient is normed. It represents the sum, minimum and maximum, and is calculated as follows: $\frac{\sum_{\forall t \in \mathcal{T}} f[t]}{\sqrt{\|\mathcal{T}\|}}$. Thus, the additive noise $Laplace(4k/\epsilon)$ is normed as well: $\frac{\sum_{\forall t \in \mathcal{T}} f[t]}{\sqrt{\|\mathcal{T}\|}} + \frac{Laplace(4k/\epsilon)}{\sqrt{\|\mathcal{T}\|}}$.

5.2.2 *Hiding Bed-Time and Wakeup-Time Hours*

According to [6], the bedtime hour is when a household switches off certain devices, e.g., the television, right before going to bed. This do not have to be the same devices for different households as long as they are usually switched off right before going to bed. We consider switch-off events only between $4pm$ and $2am$. Some appliances may still run, but only the change of consumption is of interest. An adversary trying to re-identify a household is interested in deciding whether the devices are switched off or not. Thus, an individual wants to hide the discriminative pair s_{pair} consisting of the following secrets: $s_1 =$ 'Household switches off devices before bedtime' and $s_2 =$ 'Household does not switch off devices before bedtime'. The relevant channel r includes the devices mentioned for s_{pair}.

$$f_w[x] = f_w^r[x] + f_w^1[x] + \cdots + f_w^n[x]$$

The switch-off causes a decrease of the power consumption of $0.5kWh$ on $f_w^{s_{pair}}[x]$. Thus, we apply Laplace($(4 \times 0.5)/\epsilon$) noise on Level 1 and Laplace($(4 \times \frac{0.5}{\sqrt{2}})/\epsilon$) noise on Level 2 during $4pm$ and $2am$. Hiding wakeup times is similar.

5.2.3 *Results*

It is possible to hide all other features for re-identification [6]; Table 4 lists the necessary transformations.

To quantify effectiveness, we look at the relative decrease in accuracy, i.e., the number of households re-identified with and without applying noise. While re-identification makes use of a combination of features, to isolate the effects of hiding specific secrets we only look at features relevant for the secret. While this reduces the number of households re-identified, this is the case both with and without applying noise, so our evaluation is still conclusive. We deem a household re-identified if its time series receives the n-th lowest score at least. In total, we tested 158 household from the CER data set and set $\epsilon = 0.1$. This data set consists of roughly 5000 homes in Ireland with different numbers of inhabitants, measuring electricity consumption every 30 minutes over more than one year [16]. Table 5 contains our results. It contains the feature set used for re-identification and the accuracy decrease after applying the Pufferfish framework. First, independent of the feature set, there is a significant decrease in accuracy. Thus, hiding the features in the described way is effective. However, the algorithm still can re-identify a small number of households: In our evaluation, we have assumed the same discriminative pair for all households. However, for outliers in particular, e.g., a household consuming a lot of electricity and thus being easy to re-identify, discriminative pairs should differ. In particular, the k of the interval must be larger. If the feature value of a number of households is similar, then the re-identification algorithm starts to guess. Random ‘correct’ guesses become more with $n = 5$. Still, Pufferfish allows the definition of suitable secrets to hinder re-identification. Even with secrets designed in a straightforward way without considering outliers the accuracy decreases significantly.

5.3 Utility: Welfare of a Local Energy Market

A privacy method must protect sensitive information of individuals. However, it is also important that the data can still be used for certain purposes afterwards. In order to evaluate to which extent the proposed mechanism preserves utility, we integrate it into a local energy-market scenario and measure the effect on the welfare. Welfare is a well-known and intuitively understandable economic measure. In a local energy market, consumers and producers can trade electricity. In general, this leads to a more effective allocation of renewables, including a drop of CO_2 emissions. However, individuals have to reveal their prospective consumption to other market participants. Obviously, the prospective consumption tends to be similar or even identical to the actual one. With any reasonable market mechanism, if participants reveal their true demand they will receive the highest welfare. In turn, revealing a privacy-enhanced demand induces a loss of welfare. However, protecting privacy has a value for the individuals as well. Thus it is insightful to investigate this tradeoff. This method has already been tested in another similar context, see [7] for more details.

5.3.1 Results

For our evaluation, we have studied a town with 300 persons living in households of up to five persons. The time interval examined is five days. The consumption data has come from the CER data source [16]. As renewable sources we have taken 150 photovoltaic sites as well as 150 combined heat and power plants. As privacy requirements, we have chosen to hide the bedtime and the total sum see. Since Pufferfish as well as the selection of households include randomness, we repeat each experiment ten times. We measure the relative welfare, which is the welfare using the privacy method in relation to the welfare for the original data.

Hiding the bedtime results in a welfare loss of 26% on average, with a low spread, see Figure 9. Hiding requires applying noise to 10 hours a day. This includes the consumption after $4pm$, which contains a large fraction of the daily consumption due to evening activities of households. Hiding the sum respectively the minimum and maximum consumption leads to a smaller relative welfare loss compared to the bedtime requirement on average, but has a larger spread of values. In this case, applying noise shifts all the values of the time series up- or downwards, but it keeps the shape. This is because the actual development is not influenced. Thus, we see that hiding different secrets has different effects on the utility (Figure 9). Note that the welfare loss of 26% is relative to the theoretical maximum efficiency (cf. [7]). Thus, the loss of welfare is relatively low, compared to the fraction of values modified.

5.4 Summary of Results

The evaluation has shown that Pufferfish privacy can indeed shield personal information from information-extraction approaches. The potential of an adversary to gain information from the disclosed data set has dropped significantly. On the other hand, we have shown by means of a local energy market that the utility of the resulting data set still is on an acceptable level. Again, we have used secrets that prevent state-of-the-art information-extraction methods from providing meaningful results.

6. CONCLUSIONS

Disclosure of data plays a significant role in the context of the smart grid. However, time series of smart meter data contain sensitive information, represented in many different ways. Individuals might not allow access to the data as long as sensitive information based on individual privacy preferences is not removed. Pufferfish is a state-of-the-art approach to hide specific information. However, application-specific work is required when applying it to smart meter data and carrying out an evaluation that is conclusive. This includes the definition of how sensitive information is represented, how data-evolution scenarios can be applied, and how the information can be perturbed to give Pufferfish guarantees. Next, it is challenging to evaluate the general coverage of secrets and the utility of the perturbed data. Our study has addressed these points.

Our study has featured a general way of describing secrets in smart-meter data. Transforming time series of such data is one possible way to facilitate the definition of arbitrary secrets. A certain set of transformations is sufficient to cover a broad variety of possible secrets. The precision of modern information-extraction methods then decreases significantly, which is good. On the other hand the impact on the utility of the data, measured in a real-world scenario, is tolerable.

References

[1] O. Abul, F. Bonchi, and M. Nanni. Never Walk Alone: Uncertainty for Anonymity in Moving Objects Databases. *IEEE 24th International Conference on Data Engineering*, 2008.

Feature Set	**Top** n	**w/o noise**	**noise**	**Accuracy Decr.**
Sum	1	6	2	66%
Min	1	15	1	93%
Max	1	7	3	57%
Sum,Min,Max	1	30	8	73%
Bedtime	5	8	5	37.5%
Wakeup time	5	6	3	50%
Bed-, Wakeup time	5	13	6	53.8%

Table 5: Results Re-Identification

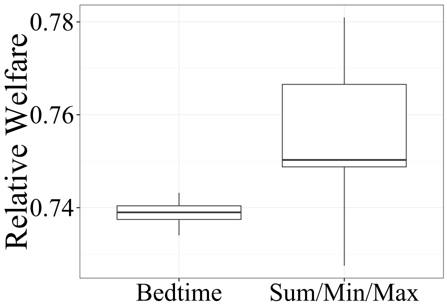

Figure 9: Relative Social Welfare for different Privacy Requirements

[2] G. Acs and C. Castelluccia. I have a DREAM! (DIffeRentially PrivatE smart Metering). *Information Hiding*, 2011.

[3] A. Albert and R. Rajagopal. Smart Meter Driven Segmentation: What Your Consumption Says About You. *IEEE Transactions on Power Systems*, 28(4), 2013.

[4] N. Batra, H. Dutta, and A. Singh. INDiC: Improved Non-intrusive Load Monitoring Using Load Division and Calibration. *12th International Conference on Machine Learning and Applications*, 2013.

[5] F. Bonchi and L. Lakshmanan. Trajectory Anonymity in Publishing Personal Mobility Data. *ACM SIGKDD Explorations*, 13(1), 2011.

[6] E. Buchmann, K. Böhm, T. Burghardt, and S. Kessler. Re-identification of Smart Meter data. *Personal and Ubiquitous Computing*, 17(4), 2012.

[7] E. Buchmann, S. Kessler, P. Jochem, and K. Böhm. The Costs of Privacy in Local Energy Markets. In *IEEE Conference on Business Informatics*, 2013.

[8] M. Burrows and D. Wheeler. A Block-sorting Lossless Data Compression Algorithm. 1994.

[9] D. M. Burton. *The History of Mathematics: An Introduction.* 6 edition, 2007.

[10] R. Coifman and M. Wickerhauser. Entropy-based Algorithms for best Basis Selection. *IEEE Transactions on Information Theory*, 38(2), 1992.

[11] A. Das. *Signal conditioning : an introduction to continuous wave communication and signal processing.* 2012.

[12] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Communications on pure and applied mathematics 41.7*, 41(7), 1988.

[13] C. Dwork. Differential privacy. *Automata, languages and programming*, 2006.

[14] H. Goncalves and A. Ocneanu. Unsupervised Disaggregation of Appliances using aggregated Consumption Data. In *The 1st KDD Workshop on Data Mining Applications in Sustainability*, 2011.

[15] G. Hart. Nonintrusive Appliance Load Monitoring. *Proceedings of the IEEE*, 80(12), 1992.

[16] Irish Social Science Data Archive. Electricity Customer Behaviour Trial. http://www.ucd.ie/issda/, 2012.

[17] G. Kalogridis, C. Efthymiou, S. Denic, T. Lewis, and R. Cepeda. Privacy for smart meters: Towards undetectable appliance load signatures. In *First IEEE International Conference on Smart Grid Communications*, 2010.

[18] S. Kessler, E. Buchmann, and K. Böhm. Deploying and Evaluating Pufferfish Privacy for Smart Meter Data (Technical Report). http://dbis.ipd.kit.edu/1724.php, 2014.

[19] D. Kifer and A. Machanavajjhala. No free Lunch in Data Privacy. *Proceedings of the International Conference on Management of Data*, 2011.

[20] D. Kifer and A. Machanavajjhala. A Rigorous and Customizable Framework for Privacy. *31st Symposium on Principles of Database Systems*, 2012.

[21] H. Kim, M. Marwah, M. Arlitt, G. Lyon, and J. Han. Unsupervised Disaggregation of low frequency Power Measurements. Number i. HP Labs Tech. Report, 2010.

[22] J. Kolter and M. Johnson. REDD: A public data set for energy disaggregation research. *Workshop on Data Mining Applications in Sustainability*, (1), 2011.

[23] S. McLaughlin, P. McDaniel, and W. Aiello. Protecting consumer privacy from electric load monitoring. In *Proceedings of the 18th ACM conference on Computer and communications security*, 2011.

[24] A. Molina-Markham and P. Shenoy. Private Memoirs of a Smart Meter. *Proceedings of the BuildSys*, 2010.

[25] M. Nergiz and M. Atzori. Towards Trajectory Anonymization: a Generalization-based Approach. *SIGSPATIAL ACM GIS*, 2(106), 2008.

[26] S. Papadimitriou, F. Li, and G. Kollios. Time series compressibility and privacy. *33rd Conference on Very Large Databases*, 2007.

[27] D. B. Percival and A. Walden. *Wavelet Methods for Time Series Analysis.* 2006.

[28] S. R. Rajagopalan, L. Sankar, S. Mohajer, and H. V. Poor. Smart meter privacy: A utility-privacy framework. *IEEE International Conference on Smart Grid Communications*, 2011.

[29] V. Rastogi and S. Nath. Differentially private aggregation of distributed time-series with transformation and encryption. In *Proceedings of the International Conference on Management of Data*, 2010.

[30] I. Richardson, M. Thomson, D. Infield, and C. Clifford. Domestic electricity use: A high-resolution energy demand model. *Energy and Buildings*, 42(10), 2010.

[31] L. Shou, X. Shang, K. Chen, G. Chen, and C. Zhang. Supporting Pattern-Preserving Anonymization For Time-Series Data. *IEEE Transactions on Knowledge and Data Engineering*, 2011.

[32] Q. Wang, V. Megalooikonomou, and C. Faloutsos. Time series analysis with multiple resolutions. *Information Systems*, 35(1), 2010.

[33] M. Wong. *Discrete Fourier Analysis.* 2011.

[34] E.-h. Yang and J. C. Kieffer. Efficient Universal Lossless Data Compression Algorithms Based on a Greedy Sequential Grammar Transform. 46(3), 2000.

[35] M. Zeifman and K. Roth. Nonintrusive Appliance Load Monitoring: Review and outlook. *IEEE Transactions Consumer Electronics*, 57(1), 2011.