Privacy Assessment of Data Flow Graphs for an Advanced Recommender System in the Smart Grid

Fabian Knirsch[1], Dominik Engel[1], Cristian Neureiter[1], Marc Frincu[2], and Viktor Prasanna[2]

[1] Josef Ressel Center for User-Centric Smart Grid Privacy, Security and Control,
Salzburg University of Applied Sciences,
Urstein Sued 1, A-5412 Puch/Salzburg, Austria
{fabian.knirsch, dominik.engel, christian.neureiter}@en-trust.at
[2] Ming-Hsieh Department of Electrical Engineering,
University of Southern California,
Los Angeles, USA
{frincu, prasanna}@usc.edu

Abstract. The smart grid paves the way to a number of novel applications that benefit a variety of stakeholders including network operators, utilities and customers as well as third party developers such as electric vehicle manufacturers. In order to roll out an integrated and connected grid that combines energy and information flows and that fosters bidirectional communications, data and information needs to exchanged and aggregated. However, collecting, transmitting and combining information from different sources has some severe privacy impacts on customers. Furthermore, customer acceptance and participation is the key to many smart grid applications such as demand response. In this paper we present (i) an approach for the model-based assessment of privacy in the smart grid that draws on a formal use case description (data flow graphs) and allows to asses the privacy impact of such use cases at early design time; and (ii) based on that assessment we introduce a recommender system for smart grid applications that allows users and vendors to make informed decisions on the deployment, use and active participation in smart grid use cases with respect to their individual privacy.

1 Introduction

In a smart grid a number of stakeholders (actors) have to cooperate effectively. Interoperability has to be assured on many layers, ranging from high level business cases to low level network communication. Data and information is sent from one actor to another in order to ensure effective communication. Furthermore, the exchange of vast amounts of data is crucial for many smart grid applications, such as demand response (DR) or electric vehicle charging [Cavoukian et al., 2010], [Langer et al., 2013]. However, this data is also related to individuals and privacy issues are an upcoming concern [McDaniel and McLaughlin, 2009], [Simmhan et al., 2011a]. Especially the combination of data, e.g., meter values and preferences for DR can exploit serious privacy threats such as the prediction of personal habits. In system engineering, privacy is a cross-cutting concern that has to be taken into account throughout the entire development life-cycle, which is also referred to as *privacy by design* [Cavoukian et al., 2010].

Model-driven privacy assessment is especially useful when applied in software engineering. In [Boehm, 2006], the author thoroughly investigates the phases in software engineering and the expected costs for error correction and change requests. Costs double with every phase and once an application or a service is delivered, the additional adding of crosscutting concerns such as privacy is tied to enormous costs. As a result, design time privacy assessment is preferred in early phases of the software engineering process. Therefore, a framework is needed to (i) model the system, including high-level use cases and concrete components and communication flows; and (ii) to assess the system's privacy impact using expert knowledge from the domain. Related work in the domain of automated assessments in the smart grid mainly focuses on security aspects and is not primarily concerned with privacy and the modeling in adherence to reference architectures.

In this paper we address these issues and present an approach for the model-driven assessment of privacy for smart grid applications. The framework proposed in this paper is designed to assist system engineers to evaluate use cases in the smart grid in an early design phase. For evaluation only meta-information is used and no concrete data is needed. We use Data Flow Graphs (DFG) to formally define use cases according to a standardized smart grid reference architecture. The assessment is based on an ontology driven approach taking into account expert knowledge from various domains, including customer views on privacy as well as system engineering concerns. The output is a set of threats and a quantitative analysis of risks, i.e., a number indicating the strength of that threat. To evaluate the system we draw on insights from the University of Southern California microgrid. The primary contributions of this paper are (i) the use of DFGs to model use cases in the smart grid; (ii) the usage of DFGs for a quantitative privacy assessment; and (iii) the use of an ontology driven approach to capture domain knowledge.

The remainder of this paper is structured as follows: In Section 2 related work in the area of smart grid reference architectures, privacy evaluation and automated assessment tools is presented. In Section 3 the architecture of the proposed framework and its components are described. This includes the concept of DFGs for modeling use cases in the smart grid, the principal design of the ontology and the mapping of data flow graphs to the ontology, the methodology for defining threat patterns and finally, how these patterns are matched to use cases. The framework is evaluated with a set of representative use cases in Section 4. Section 5 shows a practical application for the proposed framework as a recommender system for the potential privacy impact when using applications and services in the smart grid. Section 6 summarizes this paper and gives an outlook to further work in this area.

2 Related Work

In this section related work in the field of smart grid reference architectures, privacy evaluation and assessment as well as automated assessment tools are presented. Often, privacy and security are used interchangeably. For the purpose of this paper we refer to privacy as legally accessing data but not using it for the intended purpose. Security, by contrast, would involve the illegal acquisition of data. In both cases, the well established and widely understood terminology from security assessment is used, i.e., *threat*, *attacker*, *vulnerability* and *countermeasure*.

2.1 Reference Models

Stakeholders in the smart grid come from historically different areas, including electrical engineering, computer science and economics. To ensure interoperability and to foster a common understanding, standardization organizations are rolling out reference models and road maps. In the US the NIST Framework and Roadmap for Smart Grid Interoperability Standards [National Institute of Standards and Technology, 2012] and in the EU the Smart Grid Reference Architecture [CEN, Cenelec and ETSI, 2012b] were published. The European Smart Grid Architecture Model (SGAM) is based on the NIST Framework, but extends the model to better meet European requirements, such as distributed energy resources. In this paper we investigate use cases from the US. In particular we are focusing on use cases from the University of Southern California microgrid and we thoroughly discuss a typical DR use case. Investigations have, however, shown that for the purpose of this project all use cases from the US can be directly mapped to the European SGAM without the loss of information. Therefore we propose the utilization of the SGAM for two reasons: (i) the SGAM builds on the NIST model and allows to capture both, use cases from the US and the EU; and (ii) with the SGAM Toolbox [Dänekas et al., 2014] present a framework for modeling use cases based on the SGAM; in that way formally modeled use cases are the input for the evaluation.

2.2 Privacy

Privacy (and security) issues in the smart grid are addressed by standards in the US [National Institute of Standards and Technology, 2010] and the EU [CEN, Cenelec and ETSI, 2012a]. Privacy, in specific, has no clear definition. According to a thorough analysis in [Wicker and Schrader, 2011], privacy can be defined as the right of an individual's control over personal information. More formally this is defined by [Barker et al., 2009] in a four dimensional privacy taxonomy. The dimensions are *purpose*, *visibility*, *granularity* and *retention*. The *purpose* dimension refers to the intended use of data, i.e., what personal information is released for. The purpose ranges from single, a specific use only, to any. *Visibility* refers to who has permitted access. The range is from owner to all/world. *Granularity* describes to what extent information is detailed. The *retention* dimension finally is the period for storage of data. In any case, privacy is assured if all these dimensions are communicated clearly and fully disclosed to data owners and the compliance to the principles is governed. Hence, data is collected and processed for the intended purpose only, and the degree of visibility, granularity and retention is at the necessary minimum.

2.3 Assessment Tools

To measure the degree to which systems adhere to privacy requirements, approaches for automated qualitative assessments (resulting in statements of possible privacy impacts due to privacy critical actions or relationships) and quantitative assessments (resulting in a numeric value that determines the risk of privacy impacts) exist.

In [Ahmed et al., 2007], the authors present an approach towards ontology based risk assessment. The authors propose three ontologies, the *user environment ontology* capturing where users are working, i.e., software and hardware, the *project ontology* capturing concepts of project management, i.e., work packages and tasks and the *attack ontology* capturing possible attacks, e.g., non-authorized data access, virus distribution or spam emails. For a risk assessment, attacks (defined in the attack ontology) are matched with information available from the other ontologies. For a quantitative assessment, the annual loss expectancy is calculated by combining a set of harmful outcomes and the expected impact of such an outcome with the frequency of that outcome. The approach presented by Ahmed et al. is designed for security issues and does not explicitly cover privacy assessments.

In [Kost et al., 2011] and [Kost and Freytag, 2012] an ontology driven approach for privacy evaluation is presented. The aim of these papers is to integrate privacy in the design process. High-level privacy statements are matched to system specifications and implementation details. The proposed *privacy by design* process includes the following phases: identification of high-level privacy requirements, translation of abstract privacy requirements to formal privacy descriptions, realization of the requirements and modeling of the system and analyzing the system by matching formal privacy requirements to the formal system model. Contrary to our work this approach is not focused on use cases in the smart grid and therefore does not model systems based on a standardized reference architecture.

A workflow oriented security assessment is presented in [Chen et al., 2013]. This approach is not based on ontologies but on argument graphs. The presented framework uses *security goal*, *workflow and system description*, *attacker model* and *evidence* as an input. This information is aggregated in a discriminative set of argument graphs, each taking into account additional input. Nodes in the graph are aggregated using boolean expressions and the output is a quantitative assessment of the system. Instead of focusing on workflow analysis using graphs, we model systems as a whole in adherence to the standardized reference architecture using an ontology driven approach to integrate expert knowledge.

A considerably broader approach for an assessment tool that incorporates both, the balancing of privacy requirements and operational capabilities is presented in

[Knirsch et al., 2015]. This work presents a graph based approach that allows the modeling of systems with respect to the operational requirements of certain nodes (e.g. metering at a certain frequency) and the impact of privacy restrictions on subsequent nodes. The authors further present an optimum balancing algorithm, i.e. to what extent restrictions gained from privacy enhancing technologies and the necessary operational requirements can be combined. However, this needs sufficient information on how privacy is impacted by certain use cases which is provided by this work.

3 Architecture

This section is dedicated to an architectural overview as well as a detailed discussion of the components. Figure 1 shows the principal components of the proposed architecture, including input and output. For a privacy assessment, the framework accepts two inputs, a use case UC modeled as a DFG in adherence to the SGAM and a set of threat patterns T. In order to qualitatively analyze this input the use case is mapped to individuals – i.e., instances of classes – of an ontology (sometimes referred to as the *assertion box, ABox* [Shearer et al., 2008]). The corresponding class model (sometimes referred to as the *terminological box, TBox* [Shearer et al., 2008]) is based on the SGAM. This qualitative analysis provides explicit and implicit information about the elements from the DFG: actors, components, information objects and their interrelation. The results of the qualitative assessment are the input for the subsequent quantitative analysis. The output of that analysis is finally a class c from a set of classes C that the use case is assigned to. A threat pattern t is used to describe potential threats, where $t \in T$ and a class c represents a subset of threats T^*. A class c describes how threat patterns and the qualitative results are combined, which is presented as a threat matrix as an output. Note that the terminology *threat matrix* is borrowed from security analysis and that the output is not a matrix in the mathematical sense. A threat matrix compares a set of threats and the risk for these threats. Formally, the classifier is defined as Assign UC to c_i if $t \in T_i^*, \forall t \in T, 1 \leq i \leq \{C\}$. A threat exploits a set of vulnerabilities and is mitigated by a set of countermeasures. Each threat pattern can be evaluated for itself or multiple patterns are combined to classes of threats. A vulnerability is any kind of privacy impact for any kind of stakeholder or actor. Threats are evaluated using the attack vector model which is adapted from security analysis and defined in detail later in this paper. In general, an attack is feasible, if given (i) an attacker; (ii) a privacy asset; and (iii) the resources to perform the attack. Hence, a receiver or collector of privacy critical data items is potentially able to access these assets and to use them in a way not corresponding to the original purpose. This is formally represented as $\langle \text{data access}, \text{privacy asset}, \text{attack resources} \rangle$.

3.1 Data Flow Graphs

In order to qualitatively and quantitatively assess the privacy impact of a use case a formalization is crucial. In this section we introduce the concept of Data

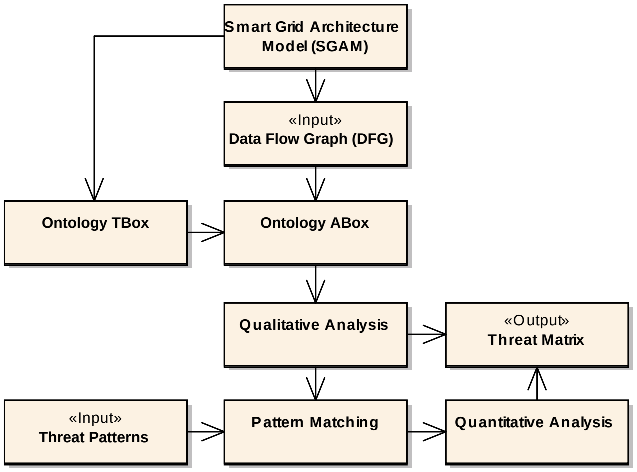

Fig. 1. Architecture overview showing input, output, components and principal information flows of the framework.

Flow Graphs (DFG) for the smart grid based on a model-driven design approach originally presented in [Dänekas et al., 2014] and [Neureiter et al., 2013]. DFGs formally capture all aspects of use cases in the smart grid in adherence to the SGAM. They contain high-level business cases as well as detailed views of a system's characteristics such as encryption and protocols. DFGs are a powerful tool as they allow both, easy modeling and full adherence to the reference architecture. Furthermore, in the graph relationships between actors, as well as the transported information objects (IO) are modeled. Nodes in a graph represent business actors, system actors or components and edges represent data flows annotated with IOs. In accordance to the standard [CEN, Cenelec and ETSI, 2012b], DFGs consist of the following five layers:

1. Business Layer. In a DFG this layer is a high level description of the business case. Business actors, their common business goal and their business requirements are modeled.
2. Function Layer. The function layer details the business case by mapping business actors to system actors and by dividing the high level business goals in use cases and steps.
3. Information Layer. This layer describes information flows in detail. System actors communicate to each other through IOs. IOs are characterized by describing information attributes on a meta-level. An IO is one of the key data used for classification and is discussed in greater detail below.
4. Communication Layer. The communication layer is a more detailed view on communication taking into account network and protocol specifications.
5. Component Layer. In a DFG this layer contains concrete components. Therefore system actors are mapped to components and devices.

Each layer is a directed graph. Both, nodes and edges can have attributes. The semantics, however, are varying. For instance, where attributed edges in the business layer describe a business case, in the information layer concrete metadata of communication flows are captured. Even though implicitly covered in the model presented above, for automated evaluation we introduce two additional layers: Between business and function layer we include the *Business Actor to System Actor Mapping* and between communication and component layer the *System Actor to Component Mapping*. This allows to capture the complexity of use cases on different levels while still maintaining the cross-layer relationship between high-level business actors and their representation as components. These layers are directed graphs as well, with edges indicating the mapping. The mapping defines a one to many relationship from business actors to system actors and from system actors to components. In the European Smart Grid Reference Architecture with the SGAM Methodology an approach for mapping use cases to the reference model is suggested. DFGs build on this methodology focusing on actors and their interrelation. An implementation for modeling DFGs in UML is available as the *SGAM Toolbox*[3]. Data Flow Graphs contain explicit information (what is modeled) and implicit information (what can be concluded). Conclusions are drawn using ontology reasoning.

3.2 Ontology Design

The ontology driven approach for classification has been chosen for two main reasons: (i) ontologies are powerful for capturing domain knowledge explicitly; and (ii) through logic reasoning [Shearer et al., 2008] ontologies are a source for implicit knowledge. The power of ontologies to formally capture knowledge and how to draw conclusions is discussed in [Guarino et al., 2009]. The power of reasoning for gaining additional, implicit knowledge can easily be outlined with two examples: In a DFG, information objects may be sent from an actor A to an actor B and from there to another actor C. This is explicitly modeled in the DFG. A reasoner in an appropriate ontology, however, may conclude directly the transitivity, hence that actor A in fact sends information to actor C. Another example is concerned with compositions of data. An information object I_1 may contain sensitive data and it may be used by an actor D to compose another information object I_2 that is sent to a collecting actor E. It is not explicitly modeled in the DFG, but it can be concluded by the reasoner, that E receives an information object which is of type sensitive data since I_2 is a composition of I_1. The ontology we propose here is designed to capture all aspects of a DFG. The ontology is modeled in OWL[4] and class expressions are stated in Manchester Syntax[5]. Therefore, all components available for modeling DFGs are represented either directly or as an abstraction in the ontology (referred to as the *TBox*). The DFG is represented in the ontology as a set of individuals (referred to as the

[3] http://www.en-trust.at/downloads/sgam-toolbox/

[4] http://www.w3.org/TR/owl-features/

[5] http://www.w3.org/TR/owl2-manchester-syntax/

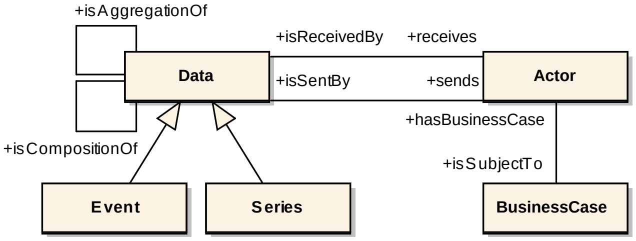

Fig. 2. Principal components of the ontology, showing a subset of the relationships between actor and data.

ABox). Figure 2 depicts the principal classes and relationships of the ontology and therefore the most relevant concepts for mapping a DFG to the ontology. This view shows the main classes and relationships for illustration purposes only; our current ontology comprises more than 60 classes, data properties and object properties. Crucial concepts represented immediately, include which actor sends or receives which data and IO and how these IOs are composed. Furthermore, a set of pre-classifiers is defined to determine implicit knowledge.

These classifiers are OWL classes using an equivalent class expression in Manchester Syntax. For instance, to determine if some aggregation consists of direct personal data, the following expression is used: `Data and isAggregationOf some DirectPersonalData`. To determine the multiplicity of the sending actor and if the data is a composition sent by many of such actors, more elaborate expressions can be phrased: `Data and isSentBy some Actor and Multiplicity value "n" and isCompositionOfMany some Data`.

3.3 Threat Patterns

In this paper we evaluate the privacy impact on customers, thus we identified the following list of typical high-level threats based on literature reviews [Cavoukian et al., 2010], [Langer et al., 2013], [Simmhan et al., 2011a]. These threats have been modified in order to be more representative for the use cases from the University of Southern California microgrid that are investigated in this paper. Subsequently, IOs that may cause these threats are determined.

Customer presence at home. This privacy concern is discussed in [Cavoukian et al., 2010]. To potentially determine a person's presence at home, some device in the customer premises is needed. This device collects data at a certain frequency, high enough to have a resolution that allows to draw conclusions on the energy usage of specific devices. Furthermore, data collected from that device needs to be sent to another actor (i.e., a utility). At the utility an individual or a system needs to have access to the data in an appropriate resolution. Since we always assume that data is accessed legally, we do not focus on unallowed data access. Additionally, the total delay of the data transmission is of relevance. If data is collected and transmitted in almost real time the presence at home can be determined immediately. If data is available with a delay only, the analysis of past events and predictions might be possible. If this information is published, an attacker might exploit this vulnerability in order to break in the house.

Tracking customer position. This threat is especially interesting for electric vehicle charging. Assuming the customer has some identification towards the charging station, at least the location, a timestamp and the amount of energy consumed will be recorded for billing. Depending on the design of the infrastructure only little information will be sent to the operator or a very detailed profile of the customer is maintained. Here, the multiplicity of the actors is crucial and the fact that different actors have access to the same data. Attacks for this threat are described in [Langer et al., 2013], e.g., using information for targeted ads, for tracking movements to certain places or to infer the income based on recharges.

3.4 Pattern Matching

Actual classification is done in the pattern matching process. For each actor in the DFG and the ontology, respectively, the attack vector is determined, i.e., to which resources does an actor have access and what is the effort. If that shows feasible matching this is seen as a threat. It can be retrieved immediately from the ontology if an actor has access to a certain IO. This is done by evaluating actor and data object properties and by incorporating information from the pre-classifiers. Furthermore, relationships on the business layer and data properties such as encryption are taken into account. The following, discriminative set of classifiers is used to determine potential threats: first, for each information object the data provider and the data collector are determined (according to the terminology defined in [Barker et al., 2009]) and it is assessed who has access to the data. This yields a list of three-tuples in the form $\langle$information object (IO), data provider (DP), data collector (DC)$\rangle$. Then it is determined if an information object either contains sensitive or direct personal data (according to the terminology defined in [The European Parliament and the Council, 1995]). This yields another three-tuple in the form $\langle$information object (IO), sensitive (S), direct personal (DP)$\rangle$. Finally it is determined if the attacker has actual data access, yielding one more three-tuples in the form $\langle$information object (IO), data collector (DC), access (A)$\rangle$. Data access depends on the relationship of actors, on data resolution, retention and encryption. Matching these tuples to each other results in the components of the attack vector, recalling $\langle$data access, privacy asset, attack resources$\rangle$ yields $\langle\langle \mathrm{IO}, \mathrm{DP}, \mathrm{DC}\rangle, \langle \mathrm{IO}, \mathrm{S}, \mathrm{DP}\rangle, \langle \mathrm{IO}, \mathrm{DC}, \mathrm{A}\rangle\rangle$. An exemplary attack vector for a DR use case where DR preferences are sent to the utility is $\langle\langle \text{DR preferences}, \text{customer}, \text{utility}\rangle, \langle \text{DR preferences}, \text{false}, \text{false}\rangle, \langle \text{DR preferences}, \text{utility}, \text{true}\rangle\rangle$. This already provides thorough qualitative analysis. It is possible to determine which actor can potentially threaten the privacy of another actor. It is even possible to conclude how and where this might happen. However, for a quantitative assessment the risk for a particular threat is calculated. While a qualitative assessment is useful in supporting detailed system design decisions and evaluation, for a very first outline of the overall system characteristics, a quantitative value is much more expressive. Further, providing a numeric value for the system's privacy impact helps to easily compare and contrast proposed designs.

Risk is calculated as the product of the *probability of occurrence* (PO) and the *expected loss* (EL). For the set T^* a number of patterns $t_{v,1} \dots t_{v,N}$ and $t_{c,1} \dots t_{c,M}$, respectively is defined. A pattern therefore contains a set of conditions for vulnerabilities $t_{v,i}$ and countermeasures $t_{c,i}$. Conditions are SPARQL ASK queries[6] that return either *true* or *false* if the pattern applies or not. For brevity, t'_v denotes the number of vulnerabilities that apply, t'_c the number of countermeasures that apply and t_v and t_c denote the total number of vulnerabilities and countermeasures, respectively. In this paper we propose the following approach for determining values for the probability of occurrence $PO(t'_v, t'_c)$ and the expected loss $EL(t'_v, t'_c)$: $PO(t'_v, t'_c)$ is determined by defining a plane that satisfies the following conditions: $PO(t'_v = t_v, t'_c = 0) = 1$, $PO(t'_v = 0, t'_c = t_c) = 0$ and $PO(t'_v = 0, t'_c = 0) = \frac{1}{2}$. This yields $PO(t'_v, t'_c) = \frac{1}{2}(\frac{t'_v}{t_v} - \frac{t'_c}{t_c} + 1)$. A linear model is chosen due to its simplicity and might be extended by more complex approaches in future. A condition that is of type *vulnerability* increases $EL(t'_v, t'_c)$, a condition of type *countermeasure* decreases $EL(t'_v, t'_c)$. The value of $EL(t'_v, t'_c)$ is defined in the pattern. Risk R is finally defined by $R = PO(t'_v, t'_c)EL(t'_v, t'_c)$.

To feed in the results gained from the qualitative analysis, certain variables in the query can be bound to instances. For example, given the following fraction of a query (where `usc` denotes the namespace prefix for actors and IOs in the University of Southern California microgrid)

```
$io usc:isSentBy ?systemactor . $io usc:isReceivedBy ?systemactor .
?systemactor usc:isRealizationOf ?businessactor .
?businessactor a usc:BusinessActor
```

to determine if *some* information object is sent by *some* business actor. It is now possible to bind the variable `$io` to a concrete value as determined in the qualitative assessment, e.g., `$io` ← `InformationObject.CustomerName`. This allows to assess a particular impact on a particular information object or component/actor based on the previously calculated attack vectors.

We developed generic patterns for *typical* threats, i.e., such as the ones mentioned above. The framework is, however, not limited to this set of patterns and allows the definition of an arbitrary number of additional patterns to meet the individual needs of the application scenario. The output of the framework is a threat matrix contrasting the results from the qualitative analysis and from the quantitative risk assessment. For a UC, a threat matrix contains the attack vector and the assigned risk for the determined class c.

For illustrative purposes, the following listing shows an example pattern for *customer presence at home*. This includes the vulnerability *device in customer*

[6] http://www.w3.org/TR/sparql11-query/

premises (exemplary assigned an EL of 4) and the countermeasure *aggregation of data from multiple customers* (exemplary assigned an EL of -6).

```
<Pattern name="customer presence at home">
  <Vulnerability
    name="device in customer premises">
    <EL>4</EL>
    <Condition>
      ?device x:isRealizationOf $ba .
      $ba a x:BusinessActor .
      ?device x:Zone
      "Customer Premises"^^xsd:string
    </Condition>
  </Vulnerability>
  <Countermeasure
    name="aggregation of data from multiple
      customers">
    <EL>-6</EL>
    <Condition>
      $io x:manyAreAggregatedBy ?io2 .
      ?io2 x:isReceivedBy ?ba1 .
      $io x:isRecevivedBy ?ba2
      FILTER (?ba1 != ?ba2)
    </Condition>
  </Countermeasure>
</Pattern>
```

4 Evaluation

For evaluating the framework new, previously unused use cases are applied. The set of threat patterns and their impact on privacy is based on the aforementioned literature reviews. We are therefore using a representative set of use cases describing typical applications in the smart grid. This includes, but is not limited to, smart metering, electric vehicle charging and DR. In this section a real-life use case from the University of Southern California microgrid, and a real-life use case from the Salzburg Smart Grid Model Region are evaluated as an example. These use cases have been chosen as they are (i) simple enough to verify results based on literature reviews; and (ii) complex enough to have an interesting combination of actors and information flows. Evaluation is performed with a prototypical implementation that uses DFGs and threat patterns as an input and produces a threat matrix as an output.

4.1 Smart Metering

For the Salzburg Smart Grid Model Region use case we investigate a typical smart metering scenario as shown in Figure 3. Smart metering is the basis for many advanced applications in the smart grid and therefore considered as a key

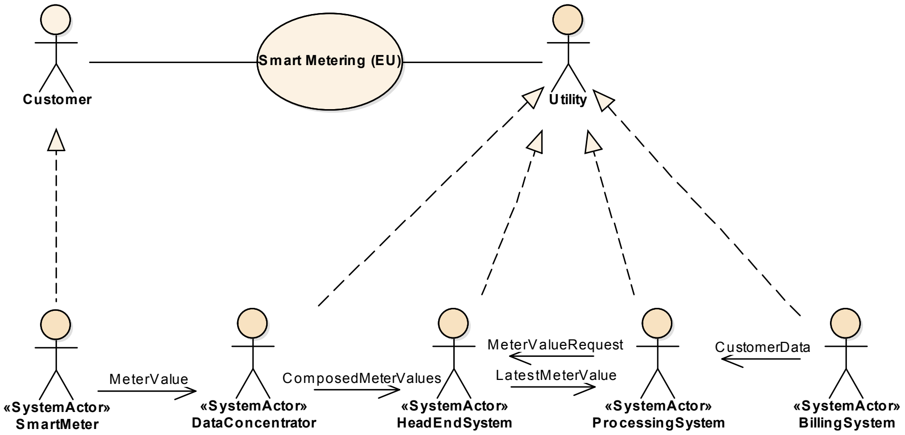

Fig. 3. Outline of the smart netering use case that is discussed for evaluation.

enabling technology [Arnold, 2011]. Today, smart metering is typically applied for network monitoring and billing. The use case is outlined as follows: once a smart meter is installed in a residential building meter values are collected at a fixed frequency. Due to regulatory provisions in this is (e.g., in Austria and Germany) limited to one value each 15 minutes and 96 values per day, respectively. Data for one day is summarized in the smart meter and forwarded to the utility on the previous day. Multiple smart meters are connected in a mash-like topology and data is sent to a data concentrator that (i) relays data from power line communication to IP; and (ii) collect data from the attached meter. Smart meter data is finally stored in a head-end system. For billing, meter data (energy consumption) is linked to additional data from the billing system, such as contract details, name, address and past payment behavior.

Actors. Business actors are the *user* and the *utility*. The user is mapped to the system actor *smart meter*. The utility is represented as *data concentrator*, *head-end system*, *billing system* and *processing system*. The latter is the component linking the data from the billing system to the data from the head-end system.

Information Objects. Meter values are sent at a fixed rate from the customer premises to the utility. The utility stores these values in the head end-system and the processing system finally combines both, data from the head-end system and data from the billing system.

Customer Presence at Home. When metering is done on a regular basis, it is easily detectable if a customer is present at home. The qualitative analysis shows that meter values are sent from the smart meter to the data concentrator and further to the head-end system. Storing in the head-end system is privacy critical, since data metered at a certain frequency is persisted. For this threat four vulnerabilities (device in customer premises, collecting data at a certain frequency, receiver has access to data, data retention is unlimited) and one countermeasure (aggregation of data from multiple customers) are identified, resulting in a *PO* of 0.9, and *EL* of 11.5 and a risk value of 10.35.

Identification of Customer Habits. While the intended use case for persisting meter data is billing, such data can be used to identify customer behavior, e.g., by running statistics and predicting future actions. For this threat eight vulnerabilities (device in customer premises, collecting data at a certain frequency, receiver has access to data, data retention is unlimited, composition of location and timestamp, different actors have access to the same data, location information with unlimited retention) and two countermeasures (aggregation of data from multiple customers, retention is for processing only) are identified, resulting in a *PO* of 0.75, an *EL* of 11.5 and a risk value of 8.63.

The model-driven assessment of the smart metering use case has shown that the risk of identifying customer habits is less than the risk of determining customer presence at home. This is due to the fact that determining presence is a yes/no decision whereas determining and predicting habits requires way more data and information.

4.2 Demand Response

For the University of Southern California microgrid use case, we are focusing on a DR scenario similar to the one described in [Simmhan et al., 2011b]. This scenario is outlined in Figure 4. A customer interested in DR creates an online profile stating on which DR actions the customer is interested to participate (e.g., turning down air condition). When the utilities want to curtail load with DR, a customer whose profile fits the current requirements is sent a text message to, e.g., turn down the air condition. This message is acknowledged by the customer and the utility further reads the meter values to track actual power reduction. Besides the data flows mentioned, this further involves the storing of the profile and the past behavior of the customer for a more accurate prediction. For modeling this use case as a DFG, the following actors and IOs are identified.

Actors. Business actors are the *user* and the *utility.* The user is mapped to the system actors *smart meter*, *device* and *portal*. DR requests are sent to the user device (e.g., a cell phone) and the user's DR preferences are set in the portal (e.g., a web service). The smart meter is used to measure actual curtailment. The utility is mapped to a *DR repository*, containing preferences for each user and past behavior, to a *prediction unit* predicting DR requests based on the preferences and a *control unit* to meter user feedback and actual curtailment.

Information Objects. Cross-domain/zone information flows include user preferences sent to the utilities, DR requests sent to the user from the utility and both, the user acknowledge/decline and the meter values sent back to the utility. Information flows within the utilities' premises are from the DR repository to the prediction unit and from the control unit to the DR repository. Given the threat patterns introduced in Section 3, we use our framework to determine the privacy impact of this use case which provides the following results.

Customer presence at home. The qualitative analysis shows that in the DR repository of the utility information about both, past customer behavior and

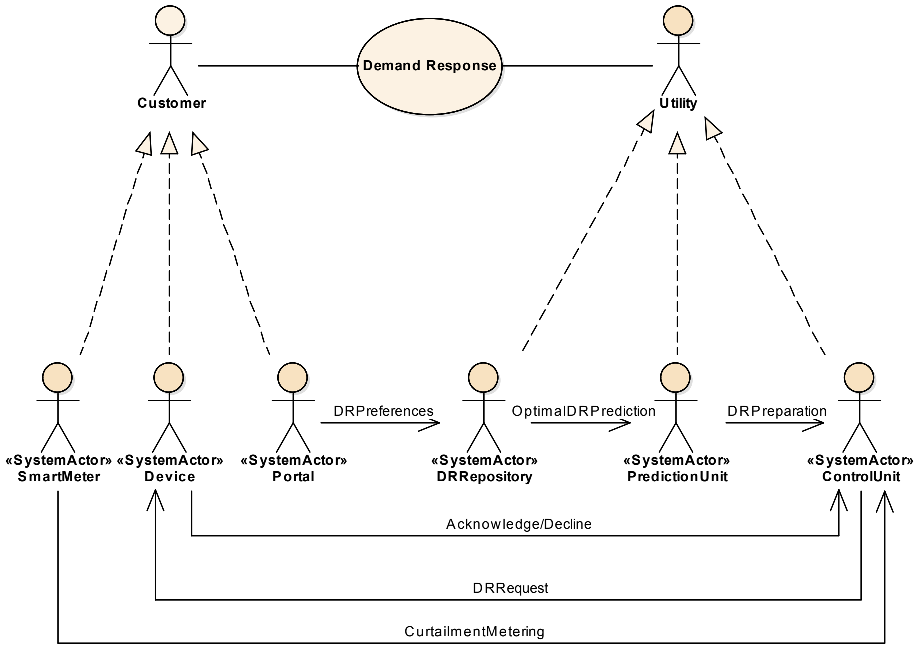

Fig. 4. Outline of the DR use case that is discussed for evaluation.

customer data is brought together, i.e., direct personal data is composed with a detailed history of a person's actions. Furthermore, the customer's acknowledge/decline and the measured curtailment reveal if a customer (i) responded to the DR request; and (ii) actually participated in DR; both is a indication for the presence at home. For this threat we identified four vulnerabilities (device in customer premises, collecting data at a certain frequency, receiver has access to data, data retention is unlimited) and one countermeasure (aggregation of data from multiple customers), resulting in a PO of 0.9, an EL of 11.5 and a risk value of 10.35.

Tracking customer position. In our case, this threat might apply in two different scenarios: First, this threat is immediate if the acknowledge/decline response to DR requests contains the customer position (e.g., if sent by a cell phone or other mobile device). This does not only show the customers past and present position, but also if the customer is able to remotely control devices in his premises. Second, when the customer is represented by an additional component *electric vehicle charging station*. Assuming that DR requests are also sent with respect to the charging behavior. Based on the amount of energy the customer is willing to DR it might be possible to estimate the consumption of the electric vehicle and subsequently the traveled distance. For this threat we identified two vulnerabilities (composition of location and timestamp, different actors have access to the same data) and one countermeasure (aggregation of data from multiple customers), resulting in a PO of 0.66, an EL of 5 and a risk value of 3.33.

The mode-driven assessment of the DR use case has shown that the risk of tracking customer position is low compared to the risk of determining customer presence at home. This result stems from the fact that there apply a number of vulnerabilities with high expected loss value, hence a device in the customer premises, data collected at a certain frequency, receiver has access to data and unlimited data retention. For the risk of detecting the customer presence at home, the same value applies as for the smart metering itself. This is due to the fact that smart meter data is used as a basis for demand response.

5 Recommender System

Having a framework for assessing the privacy impact of a use case in the smart grid is a powerful foundation for building a recommender system. The objective of such a recommender system is to provide users with the ability to decide on the usage of certain application and services in the smart grid based on the privacy impact of these applications and services. We therefore adapt the policy decision point (PDP) and policy enforcement point (PEP) patterns for a recommender system as originally presented in [Knirsch, 2014]. This is primarily targeting users in order to allow them having full control over information flows, but also the utilities and the vendors of third party applications.

The principal PDP-PEP architecture is standardized as Extensible Access Control Markup Language (XACML) in [Rissanen, 2013]. This architecture has already been applied to the smart grid by Jung et al. in [Jung et al., 2012]. The recommender system we present here enhances this approach by enabling an automated assessment of applications and use cases, respectively.

In general, a PDP is a component that evaluates access requests and issues some authorization. The PDP therefore provides some mechanisms to authenticate users, usually by prompting credentials such as username and password. The PDP then checks in a repository (policy store) if a certain user is granted access to a certain resource. The assessment framework presented in the previous sections is used as a PDP in order to allow privacy-aware data retrieval in the smart grid. The scenario at hand is as follows: a user wants to access a new application or service in the smart grid. This application or service has a certain privacy impact that has been assessed with this framework upon registration (*Registration of Application*). Additionally, the application or service is governed by a PEP. The PEP redirects the user to a PDP that displays to the user a list of privacy implications associated with this particular application or service. The user is then requested to confirm the intention to use the application or service. If the user accepts, the PEP grants access (*Accessing Applications*). For this system, we propose a traffic light-styled display of privacy implications (red: high risk, yellow: medium risk, green: no or low risk) with the option to show the full, detailed analysis.

Registration of Application:

1. A vendor submits a formal application description (DFG) to the recommender system.
2. The recommender system performs the model-based privacy assessment; this yields a set of qualitative metrics (attack vectors) that are stored in the PDP.

Accessing Applications:

1. A user request access to a new application registered at the recommender system.
2. The PEP of this application checks if the user has already allowed access.
3. If *no*, the user is redirected to the PDP and the qualitative assessment is performed based on the user's role (i.e., which business actor corresponds the user to for variably binding and business actor and information object) and the user allows or denies access.
4. If *yes*, the user is forwarded to the application.

In our prototypical implementation as presented in [Knirsch, 2014], Java 1.7 Servlets running on Apache Tomcat 7 represent PDP and PEP, respectively. A user request for an application is guarded by a PEP and forwarded to the PDP, including information about the intended application and the sending party. The PDP performs an ontology driven privacy assessment for the particular use case with a predefined set of threat patterns and displays the result to the user. The result shown includes (i) a summary for the overall privacy impact (traffic light: high, medium, no or low) in appropriate colors for immediate recognizability; and (ii) an optional detailed view showing the full threat matrix. The user is requested to either continue and allow access or cancel. If the user decides to continue, the browser is forwarded to the application. In case the user cancels, one is directed back to the PEP which displays that access will not be granted. For the prototypical implementation, the set of applications is given by the use cases defined above. As the focus is on demonstrating the PDP-PEP pattern for ontology-driven privacy assessment there is no actual implementation of the use cases, i.e., no application that actually performs demand response or the like. In practical use the formal use case description will be provided by either third-parties or the providers of the application themselves.

6 Conclusion and Future Work

In this paper we introduced both, a framework for the model-driven privacy assessment in the smart grid and an advanced recommender system based on that framework. The framework itself builds on an ontology driven approach matching threat patterns to use cases that are modeled in adherence to standardized reference architectures. The approach presented here builds on meta-information and high-level data flows. It has been shown how to utilize this framework to successfully assess the privacy impact on use cases in early design time. Exemplary threats and exemplary use cases draw on insights from the University of Southern California microgrid. Further we proposed a recommender system based on the PDP-PEP pattern. This system utilizes our privacy assessment framework in order to provide users the option to allow or deny access to applications and services based on their privacy impact. Future work will include the rolling out of our recommender system to a real-world setting.

Acknowledgment

The financial support of the Josef Ressel Center by the Austrian Federal Ministry of Economy, Family and Youth and the Austrian National Foundation for Research, Technology and Development is gratefully acknowledged. Funding by the Austrian Marshall Plan Foundation is gratefully acknowledged. The authors would like to thank Norbert Egger for his contribution to the prototypical implementation. Funding by the Federal State of Salzburg is gratefully acknowledged.

This material is based upon work supported by the United States Department of Energy under Award Number number DE-OE0000192, and the Los Angeles Department of Water and Power (LA DWP). The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof, the LA DWP, nor any of their employees.

References

[Ahmed et al., 2007] Ahmed, M., Anjomshoaa, A., Nguyen, T., and Tjoa, A. (2007). Towards an ontology-based risk assessment in collaborative environment using the semanticlife. In *Proceedings of the The Second International Conference on Availability, Reliability and Security*, ARES 07, pages 400–407, Washington, DC, USA. IEEE Computer Society.

[Arnold, 2011] Arnold, G. (2011). *Green IT: Technologies and Applications*. Springer Berlin Heidelberg.

[Barker et al., 2009] Barker, K., Askari, M., Banerjee, M., Ghazinour, K., Mackas, B., Majedi, M., Pun, S., and Williams, A. (2009). A data privacy taxonomy. In *Proceedings of the 26th British National Conference on Databases: Dataspace: The Final Frontier*, BNCOD 26, pages 42–54, Berlin, Heidelberg. Springer.

[Boehm, 2006] Boehm, B. (2006). A view of 20th and 21st century software engineering. In *Proceedings of the 28th International Conference on Software Engineering*, ICSE 2006, pages 12–29, New York, NY, USA. ACM.

[Cavoukian et al., 2010] Cavoukian, A., Polonetsky, J., and Wolf, C. (2010). Smartprivacy for the smart grid: embedding privacy into the design of electricity conservation. *Identity in the Information Society*, 3(2):275–294.

[CEN, Cenelec and ETSI, 2012a] CEN, Cenelec and ETSI (2012a). Smart Grid Information Security. Technical report, CEN/Cenelec/ETSI Smart Grid Coordination Group Std.

[CEN, Cenelec and ETSI, 2012b] CEN, Cenelec and ETSI (2012b). Smart Grid Reference Architecture. Technical report, CEN/Cenelec/ETSI Smart Grid Coordination Group Std.

[Chen et al., 2013] Chen, B., Kalbarczyk, Z., Nicol, D., Sanders, W., Tan, R., Temple, W., Tippenhauer, N., Vu, A., and Yau, D. (2013). Go with the flow: Toward workflow-oriented security assessment. In *Proceedings of New Security Paradigm Workshop (NSPW)*, Banff, Canada.

[Dänekas et al., 2014] Dänekas, C., Neureiter, C., Rohjans, S., Uslar, M., and Engel, D. (2014). Towards a model-driven-architecture process for smart grid projects. In Benghozi, P.-J., Krob, D., Lonjon, A., and Panetto, H., editors, *Digital Enterprise Design & Management*, volume 261 of *Advances in Intelligent Systems and Computing*, pages 47–58. Springer International Publishing.

[Guarino et al., 2009] Guarino, N., Oberle, D., and Staab, S. (2009). *What Is an Ontology?* Handbook on Ontologies – International Handbooks on Information Systems. Springer, 2nd edition.

[Jung et al., 2012] Jung, M., Hofer, T., Dbelt, S., Kienesberger, G., Judex, F., and Kastner, W. (2012). Access control for a smart grid SOA. In *Proceedings of the 7th IEEE Conference for Internet Technology and Secured Transactions*, pages 281–287, London, UK. IEEE.

[Knirsch, 2014] Knirsch, F. (2014). Model-driven Privacy Assessment in the Smart Grid. Master's thesis, Salzburg University of Applied Sciences.

[Knirsch et al., 2015] Knirsch, F., Engel, D., Frincu, M., and Prasanna, V. (2015). Model-based assessment for balancing privacy requirements and operational capabilities in the smart grid. In *Proceedings of the 6th Conference on Innovative Smart Grid Technologies (ISGT2015)*. to appear.

[Kost and Freytag, 2012] Kost, M. and Freytag, J.-C. (2012). Privacy analysis using ontologies. In *CODASPY '12 Proceedings of the second ACM conference on Data and Application Security and Privacy*, pages 205–2016, San Antonio, Texas, USA. ACM.

[Kost et al., 2011] Kost, M., Freytag, J.-C., Kargl, F., and Kung, A. (2011). Privacy verification using ontologies. In *Proceedings of the 2011 Sixth International Conference on Availability, Reliability and Security*, ARES '11, pages 627–632, Washington, DC, USA. IEEE Computer Society.

[Langer et al., 2013] Langer, L., Skopik, F., Kienesberger, G., and Li, Q. (2013). Privacy issues of smart e-mobility. In *Industrial Electronics Society, IECON 2013 - 39th Annual Conference of the IEEE*, pages 6682–6687.

[McDaniel and McLaughlin, 2009] McDaniel, P. and McLaughlin, S. (2009). Security and privacy challenges in the smart grid. *Security Privacy, IEEE*, 7(3):75–77.

[National Institute of Standards and Technology, 2010] National Institute of Standards and Technology (2010). Guidelines for smart grid cyber security: Vol. 2, privacy and the smart grid. Technical report, The Smart Grid Interoperability Panel – Cyber Security Working Group.

[National Institute of Standards and Technology, 2012] National Institute of Standards and Technology (2012). NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 2.0. Technical Report NIST Special Publication 1108R2, National Institute of Standards and Technology.

[Neureiter et al., 2013] Neureiter, C., Eibl, G., Veichtlbauer, A., and Engel, D. (2013). Towards a framework for engineering smart-grid-speficic privacy requirements. In *Proc. IEEE IECON 2013, Special Session on Energy Informatics*, Vienna, Austria. IEEE.

[Rissanen, 2013] Rissanen, E. (2013). eXtensible Access Control Markup Language (XACML) Version 3.0. Online. http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf.

[Shearer et al., 2008] Shearer, R., Motik, B., and Horrocks, I. (2008). Hermit: A highly-efficient owl reasoner. In Dolbear, C., Ruttenberg, A., and Sattler, U., editors, *OWLED*, volume 432 of *CEUR Workshop Proceedings*. CEUR-WS.org.

[Simmhan et al., 2011a] Simmhan, Y., Kumbhare, A., Cao, B., and Prasanna, V. (2011a). An analysis of security and privacy issues in smart grid software architectures on clouds. In *IEEE International Conference on Cloud Computing (CLOUD), 2011*, pages 582–589. IEEE.

[Simmhan et al., 2011b] Simmhan, Y., Zhou, Q., and Prasanna, V. (2011b). Semantic information integration for smart grid applications. In Kim, J. H. and Lee, M. J., editors, *Green IT: Technologies and Applications*, pages 361–380. Springer, Berlin Heidelberg, Germany.

[The European Parliament and the Council, 1995] The European Parliament and the Council (1995). Official Journal L 281, 23/11/1995 P. 0031 - 0050 – Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995. Online.

[Wicker and Schrader, 2011] Wicker, S. and Schrader, D. (2011). Privacy-aware design principles for information networks. *Proceedings of the IEEE*, 99(2):330–350.