A Threat Analysis Methodology for Smart Home Scenarios

Kristian Beckers[1(✉)], Stephan Faßbender[1], Maritta Heisel[1], and Santiago Suppan[2]

[1] paluno - The Ruhr Institute for Software Technology – University of Duisburg-Essen, Essen, Germany
{kristian.beckers,stephan.fassbender, maritta.heisel}@paluno.uni-due.de
[2] Siemens AG, Munich, Germany
santiago.suppan.ext@siemens.com

Abstract. A smart grid is envisioned to enable a more economic, environmental friendly, sustainable and reliable supply of energy. But significant security concerns have to be addressed for the smart grid, dangers range from threatened availability of energy, to threats of customer privacy. This paper presents a structured method for identifying security threats in the smart home scenario and in particular for analyzing their severity and relevance. The method is able to unveil also new threats, not discussed in the literature before. The smart home scenario is represented by a context-pattern, which is a specific kind of pattern for the elicitation of domain knowledge [1]. Hence, by exchanging the smart home pattern by a context-pattern for another domain, e.g., clouds, our method can be used for these other domains, as well. The proposal is based on Microsoft's Security Development Lifecycle (SDL) [2], which uses Data Flow diagrams, but proposes new alternatives for scenario definition and asset identification based on context-patterns. These alleviate the lack of scalability of the SDL. In addition, we present Attack Path DFDs, that show how an attacker can compromise the system.

Keywords: Smart grid · Attack pattern · Threat analysis · Requirements engineering · Context

1 Introduction

A smart grid provides energy on demand from distributed generation stations of energy suppliers to prosumers that buy energy and also sell small amounts of energy. Prosumers live in smart homes, which use information technology to control smart appliances, e.g., heaters via end points such as smart phones. This is

Part of this work is funded by the German Research Foundation (DFG) under grant number HE3322/4-2 and the EU project Network of Excellence on Engineering Secure Future Internet Software Services and Systems (NESSoS, ICT-2009.1.4 Trustworthy ICT, Grant No. 256980).

J. Cuellar (Ed.): SmartGridSec 2014, LNCS 8448, pp. 94–124, 2014.
DOI: 10.1007/978-3-319-10329-7_7

one possible example of the two-way communication between technical elements ans stakeholders, such as the prosumers, his/her smart appliances, energy suppliers, etc., which the smart grid relies on. Significant security concerns have to be addressed for smart grids, due to the possible dangers of missing availability of energy for customers, as well as threats to the integrity and confidentiality of customer's data. These concerns are of particular relevance, because energy grids have a significantly longer lifespan than, e.g., telecommunication networks [3]. In addition, privacy threats, e.g., the possibility of creating behavioral profiles of prosumers, if their energy consumption data is transmitted over the grid in small time intervals [4]. These concerns have been analyzed by several organizations such as NIST [5] and even tools for penetration testing of Smart Meters exist[1].

However, all of these analyses investigate either the entire grid or focus on one particular element, e.g., a Smart Meter. We present a focused threat analysis for the smart home scenario in particular, because it is vital for the acceptance of smart grids to show the Prosumer that a secure operation of the grid is possible. A report from the security darkreading blog[2] states that the smart grid vendor Itron in the U.S., as well as the MidAmerican Energy Company have made the Microsoft's Security Development Lifecycle (SDL)[3] mandatory for the development of all software products. Furthermore, the government of India endorses practices of the SDL. Thus we rely on Microsoft's SDL in our analysis, as one of the best known security-development-life-cycle methodologies [6]. This will facilitate the adoption of our method among software requirements engineers. From a security perspective, Microsoft's SDL is very thorough in architectural threat analysis [7] and thus, recommended [8] and sometimes mandatory, as mentioned above. In particular, we improve the threat analysis of the SDL with a pattern based description for scenarios and refine some of its steps. Our contributions are a specific context-pattern for the smart home scenario that can be instantiated for any smart home scenario and re-use the results of our threat analysis. Our smart home context-pattern helps to elicit domain knowledge by describing common structures, stakeholders, and their relations. In [1], we described our smart home pattern which is based on smart grid context descriptions of standards and technical documents, and the experience of the industrial partners of the NESSoS[4] project. The usage of our smart home context-pattern has several benefits in comparison to the textual approach of Microsoft's SDL. The information about the scenario can be captured in a structured way by instantiating all elements of the pattern. The instantiation can be checked for completeness automatically and for soundness by a domain expert. The graphical representation

[1] The termineter homepage: https://code.google.com/p/termineter/ (last visited on: 8-1-2014).

[2] A report from the darkreading security blog: http://www.darkreading.com/applications/scadasmart-grid-vendor-adopts-microsofts/240000526?itc=edit_in_body_cross (last visited on: 8-1-2014).

[3] Note the SDL is an evolving concept even at Microsoft, but for simplicity's sake we consider only the SDL described in [2] for the remainder of this paper.

[4] The Network of Excellence on Engineering Secure Future Internet Software Services and Systems (NESSoS) homepage: http://www.nessos-project.eu.

of all elements helps to elicit external dependencies by analyzing the relations in the pattern. The graphical pattern helps also to discuss with the stakeholders if an element of the scenario is missing.

We aim to improve the SDL's threat analysis via turning it into a completely model-based method, meaning that every step of the method relies on models. Models are an abstraction of reality and contain relevant parts for our threat analysis. Models allow us to iterate over the elements and answer certain questions such as if an element presents value to the customer of the threat analysis, meaning: is it an asset? In addition, models help us to achieve completeness of a threat analysis, because we can check if all elements are considered or not. However, if an element is missing in the model, the threat analysis will not consider it. In order to prevent the threat analysis from analyzing an incomplete data flow diagram (abbreviated: "DFD"), we propose to use a model for the initial steps (steps 1 to 4 see Sect. 2.2) of the SDL threat analysis, as well. In particular, we propose to use the smart home context-pattern introduced previously. The information in the smart home pattern can be mapped to a DFD with little effort. Furthermore, the smart home pattern contains structural information and the DFD refines this information with data flows of the scenario. This information is vital for the threat analysis of smart home systems, because a major security issue is to restrict the flow of energy consumption data. The reason is that energy consumption data is considered personal information, as behavioral profiles can be derived from it, e.g., when inhabitants take a shower.

A fundamental difference between Microsoft's SDL and our method is that we do not categorize every element of a DFD as an asset. We define assets as everything that has value to a stakeholder in the scope of the analysis. We consider elements outside the scope, e.g., for external dependencies. Moreover, we analyze threats by identifying assets an attacker wants to harm, identify entry points of the attacker, identify vulnerabilities the attacker can exploit and define attack paths from entry points to assets. The attack paths are modeled in specific DFDs that show the data flows caused by a certain attacker type, e.g., network attacker from all entry points to the assets in so-called *attack path DFDs*.

Moreover, our threat analysis methodology is based on (1) context-pattern for model-based, high level, and re-usable scenario description and (2) DFDs for design level analysis. We assume that these basis of our methodology can be adapted to other security development lifecycle approaches such as the Comprehensive, Lightweight Application Security Process (CLASP) by the Open Web Application Security Project (OWASP) [9], as well. CLASP contains definitions of process phases. In particular, CLASP contains one phase called *Perform security analysis of system requirements and design (threat modeling)*. The input for this phase are security, business, and functional requirements, while the output of this phase are documented system threats, refined security requirements, and an architectural impact analysis. We can imagine that the security, business, and functional requirements can each refer to elements of the smart home pattern to ensure that their statements refer to the smart home scenario. Our mapping

from the smart home pattern to the DFDs can be used to analyse and describe threats in relation to the architecture. Hence, we assume that our methodology can be adapted to other security development lifecycle approaches.

The remainder of the paper is organised as follows. Section 2 presents background knowledge on smart grids, and Microsoft's SDL, and discusses the difference of our research to the related work. Section 3 describes our structured threat analysis method. Section 4 shows an example application of our method to a industrial smart home scenario. Finally, Sect. 5 concludes this work. In addition, we present an extended version of this paper in a technical report, which is available for the interested reader[5].

2 Background and Related Work

We introduce background on smart grids in Sect. 2.1, describe Microsoft's security development lifecycle in Sect. 2.2, and discuss related work in Sect. 2.3.

2.1 Background on Smart Grids

Based on the definitions of the European Commission [10], the European Smart Grid Task Force[6], and the Office of Electricity Transmission and Distribution[7], the smart grid can be described as a large, flexible, self-monitoring, self-balancing, and self-regulating electricity infrastructure which uses two-way digital communication to gather and respond to information in an automated manner in order to improve the efficiency, reliability (meaning safety and security), and sustainability of the production and distribution of energy. This new infrastructure will be able to efficiently integrate the behavior and actions of all users connected to it. This means generators, consumers, those that do both, and other third parties that provide services besides energy generation.

The European Network and Information Security Agency provides a brief overview of basic ICT components, which are: (i) operational systems, (ii) classic IT systems, (iii) communication and network protocols and (iv) end points. Each of these components has well known security threats, which facilitate to identify their possible weaknesses in the future electrical grid. However, the combination of these components and their interaction will create further, yet unknown security issues. In a smart grid every stakeholder will have the capability to remotely interact with every component of the grid, in an authorized or in a maliciously way. Security of the smart home and its information assets will prove to be critical for the grid's security. For example, Smart Meter measurements is the key information on which automated energy load estimation is based on. If data integrity is comprised and meter measurements are changed, energy supply

[5] Technical report: http://www.uml4pf.org/publications/smarthome.pdf.

[6] http://ec.europa.eu/energy/gas_electricity/smartgrids/taskforce_en.htm (last visited on 15-12-2013).

[7] http://energy.gov/oe/technology-development/smart-grid (last visited on 15-12-2013).

switch offs of a house or a sector could happen, for safety reasons, if one or several compromised meters report a dangerously high consumption rate [11].

2.2 Threat Analysis in Microsoft's Security Development Lifecycle

We propose a threat analysis based on the Microsoft Security Development Lifecycle (SDL) [2], because of its widespread application. Threat analysis is part of the risk analysis stage of the SDL and consists of the following steps, which concern a software that we call System-under-Analysis (SuA):

(1) **Define use scenarios** to identify all relevant information about the scenarios in which the SuA is used, e.g., types of stakeholders and to define key threat scenarios, e.g., theft of a device or insider threat scenarios.
(2) **Gather a list of external dependencies** means to identify essential software and hardware elements on which the SuA depends e.g. an operating system or a database.
(3) **Define security assumptions** about the environment in which the SuA is located. The environment means the elements of the external dependencies and further elements defined in the scope. An assumptions could be, that databases stores authentication information in an encrypted way.
(4) **Create external security notes** that constrain stakeholders or technical elements that interact with the SuA, e.g., only an IT administrator is allowed to change the configuration of the SuA.
(5) **Create one or more data flow diagrams (DFDs) of the application being modeled**, which is the SuA and its environment is modeled in DFDs (see Table 5 for an overview on DFD elements). The DFD with the highest abstraction level is called the context diagram. Complex processes of the context diagram are refined in separate DFDs.
(6) **Determine threat types** by using the STRIDE threat taxonomy [2]. STRIDE categorizes different actions conducted by an attacker. These actions are assigned to DFD elements defined in Step (5).
(7) **Identify the threats to the system** by listing all DFD elements. Howard and Lipner [2] simply define all DFD elements as assets. Complex processes can be refined in further DFDs. In this case, the processes in the refined DFDs are the assets and not the complex processes. Note that data flows connected to a complex process are always assets.
(8) **Determine risk** with a risk level from 1 to 4, with risk level 1 being the highest. Risks are the chance of an attack multiplied with its damage potential. All threats are labeled with a risk level depending on the chance of an attack and the potential damages. An exception are repudiation threats that are difficult to assess, because they refer to actions that are not noticed. The authors state that these risks are usually assigned the risk level of a corresponding tampering threat.
(9) **Plan mitigation** refers to the possible mitigations of risks and proposes the following mitigation strategies: *do nothing*, *remove the feature*, *turn off the feature*, *warn the user*, and *counter the threat with technology*.

2.3 Related Work

Related work on threats affecting the smart grid exist, but is often too general, as the whole smart grid information network is the scope of the threat analysis, which includes several stakeholders, and technologies. The following list of related work provides an overview and outlines structural benefits for our subsequent work, but also drawbacks from generalization or high level descriptions.

The Public Interest Energy Research Program (PIER)[8] is a project report on smart grid cyber security. The report describes threats for the smart grid. The reported security issues are derived from Wikipedia and the Open Smart Grid shared documents. There is a total number of 26 threats listed (page 26) and mapped to 9 smart grid security issues, security goals and threat levels. The result is a mostly general overview, which neither employs a clear methodology for threat derivation, nor provides concrete information on the endangered assets and therefore, cannot be used as basis for requirements elicitation.

The European Network and Information Security Agency (ENISA) provides in the annex to their smart grid report insight on ICT components and vulnerabilities in the smart grid[9]. A threat classification is given, which comprises: (1) accidental/inadvertent threats, which can be divided into (2) safety failures, (3) equipment failures, (4) carelessness, (5) natural disasters and (6) deliberate threats. Several threats are subsequently assigned to the threat classes in form of an overview table, but it remains unclear why these threats were chosen and why they are assigned to each class. The document neither provides further description on the classification, nor does it link to the source of threat identification. The incomprehensible classification of several threats, e.g., "propaganda" as a "technical threat", hinder the use of its threat catalog for future work.

Aloul et al. survey literature on smart grid complexity, vulnerabilities, attacks and proposed solutions [3]. Their work is based on the smart grid architecture proposed by the National Institute of Standards and Technology (NIST). The authors conduct a threat and attacker analysis. However, attacks are only briefly related to vulnerable ICT components, but without addressing the smart grid architecture presented previously. As a result, vulnerabilities and attacks cannot be linked to our scenario directly considered in this work. scenarios as well.

Wang et al. detail cyber security threats and requirements related to high-level "security objectives" [12, p. 1348], which is the CIA-triad. Wang et al. use well know technologies and metrics from the Internet as a comparison, and derive threats and requirements according to the security protection goals of the triad. Future work can profit from their structured approach, although the authors themselves describe the results at high and non-technical level. In addition, Yang et al. introduce a graphical impact analysis model for the smart grid.

[8] http://www.energy.ca.gov/2012publications/CEC-500-2012-047/CEC-500-2012-047.pdf (last visited on 15-12-2013).

[9] https://www.enisa.europa.eu/activities/Resilience-and-CIIP/critical-infrastructure-and-services/smart-grids-and-smart-metering/ENISA_Annex%20II%20-%20Security%20Aspects%20of%20Smart%20Grid.pdf (last visited on 15-12-2013).

Yang et al. [13] apart form the model description, general aspects considering threats and requirements can be found. In the future, the proposed impact analysis can be used subsequently after our structured threat elicitation. It will be useful when we broaden our scope, but as it is defined right now it does not concern the details of our smart home scenario. Moreover, McDaniel et al. [14] give a high-level introduction on security and privacy challenges. But they highlight and discuss the challenges without going into detail. Thus, the work is not providing any foundation for future work.

SINTEF [15] surveys and analyses security threats associated with the deployment of an Advanced Metering Infrastructure (AMI) in the Demo Steinkjer demonstration project. The derived threats focus on energy supplier communication. The method SINTEF uses is also based on Microsoft's SDL, which provides a complementary view of threats outside the scope of this paper. In addition, the authors enlist vulnerabilities based on a DFD and afterwards identify assets and draw attack trees for attacker goals such as "Compromise meter". In contrast, our method identifies assets first and focuses our threat analysis on modeling attacker behavior via identifying possible entry points and identifying vulnerabilities that can be exploited to harm the assets.

Dhillon [16] models the flow of information in a system and investigates possible interaction points of an attacker with the system. The author proposes to use annotations on the models for security relevant information, e.g., authentication data flows. These annotations are used to check, for example, that a database is the entry point for possible threats. These annotations can complement our work in the future and improve the vulnerability analysis. However, this work is not specific to smart grids.

3 A structured Method for Smart Grid Threat Analysis

We show our method in Fig. 1 and explain it in the following. For simplicity's sake, we do not consider the determination risk and plan mitigation steps in our method.

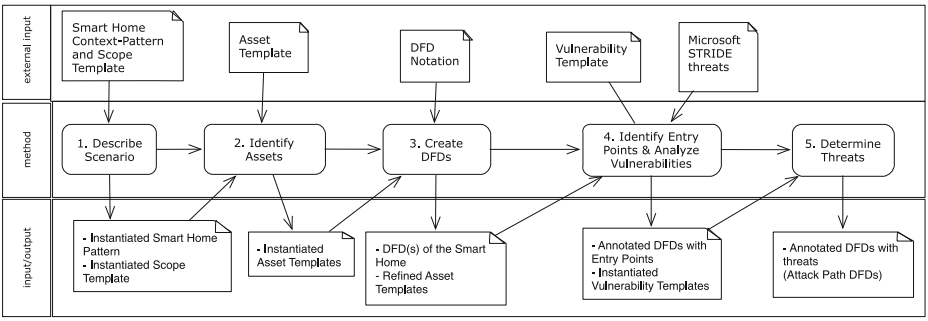


Fig. 1. A Structured Method for Smart Grid Threat Analysis

Step 1. Describe Scenario - The scenario description shall include all relevant elements of the scope and its environment. The description begins with the scope and focus of the analysis, modeling the target of analysis at an adequate level of abstraction, identifying stakeholders and relevant technical elements. Afterwards, stakeholders and relevant technical elements outside the scope (in the environment) are determined. A thorough description of the environment is essential, because stakeholders and technical elements in the environment can be external dependencies for assets (c.f., Sect. 4) in the scope of the analysis. We base these descriptions on the smart home pattern (c.f., Sect. 4) and instantiate it for the particular smart home scenario. Meaning all stakeholders and technical elements have to be labeled with the particular names in the particular scenario. The pattern can be extended with further stakeholders and technical elements for a particular scenario. If these extensions appear in multiple instantiations (scenarios), a discussion should decide if these elements shall even become part of the pattern itself. The pattern is accompanied by a scope template (c.f., Sect. 4). This template lists the elements of the scope and the elements that are not part of it, and a reasoning *why* they are left out. Moreover, we use the instantiated indirect environment of the smart home pattern to consider relevant laws and regulations. We list the relevant legal demands in the scope template.

Step 2. Identify Assets - The Microsoft SDL is lacking a precise definition of an asset. Thus, we use the definition of the ISO 27001 standard. The ISO 27001 standard defines an asset [17, p. 2] as follows: "anything that has value to the organization". The organization in our case are the stakeholders in the scope of our analysis. We identify assets in the smart home pattern by analyzing the instantiated scope template and the instantiated smart home pattern. The associations (vertices between the stakeholders) in the scope are a starting point. We check if the elements at the end of the associations potentially have value to the stakeholders and, thus, are assets. We describe the assets in asset templates. For each asset, we have to define external dependencies. The analysis of the external dependencies leads to security assumptions and to security notes for the environment. The asset templates are refined during method each time further information, e.g., due to refinement of scope elements, becomes available. Key threat scenarios are also considered in the SDL to conclude this step, but their additional benefit is left unclear compared to the effort of their identification. Thus, key threat scenarios are omitted in this method.

Step 3. Create DFDs - At this point, we have described the scenario and identified the assets of our threat analysis. We base our threat analysis on DFDs as proposed by the SDL. In addition, DFDs help to refine the technical details in the smart home pattern. Hence, we need to map the smart home pattern elements to the DFDs. Note that we only map elements of the direct environment of the pattern (see Fig. 2), because the indirect environment only contains laws and regulations, which have been considered in the first step. We map the domain knowledge in the smart home pattern to DFDs (c.f., Sect. 4). Moreover, we have more details in the DFD than in the smart home pattern. Hence, we refine the asset templates with additional information and if necessary instantiate further asset templates.

Step 4. Identify Entry Points and Analyze Vulnerabilities - The next step is to model the attackers. In particular, this step conducts an identification of possible entry points of an attacker as suggested by [18]. We elicit possible *entry points* (c.f., Sect. 4) of attackers that want to harm the previously identified assets. We suggest to use basic attacker types as proposed in our previous work [19,20]: *Physical Attacker*s threaten the physical elements of the system, e.g., hardware or buildings that host computers; *Network Attacker*s threaten *network connections* within the target of analysis; *Software Attacker*s threaten software components of the system, e.g., the application configuration inside the *Smart Meter*; *Social Engineering Attacker*s threaten humans, e.g., Prosumers[10]. We specify all possible entry points for an attacker in an annotated DFD. The DFD contains a symbol of a red triangle with an exclamation mark in the middle to illustrate the entry points (see Fig. 6). We use the previously defined entry points and specify concrete threats for each entry point using the STRIDE threat taxonomy. We use a vulnerability template to document the STRIDE threat type, attacker type, and a description of a possible exploit. Section 4 provides examples for these activities.

Step 5. Determine Threats - We use the entry points to elicit *attack paths*, which are based on Microsoft Threat Modeling. An attack path is a description of an attack from an entry point to an asset [18]. Hence, we propose so-called *Attack Path DFD*s to describe threats an attacker possibly causes towards an asset. These are DFD diagrams with an attacker process that illustrate possible ways from all entry points the attacker can use to arrive at the location of the asset and ways to harm it. The diagram is created by trying to reach one asset from all entry points. All relevant entry points and all relevant elements from the previous DFDs are part of an *Attack Path DFD*. It is also possible to exclude an entry point for an asset via reasoning. For example, if there is no path using data flows from an entry point to an asset, that entry point is not relevant for that asset. The possible use of exploits documented in the previous step are modeled in the DFD, as well (see Sect. 4 for details). The *Attack Path DFD*s are used to discuss and document the relevant threats towards the system-to-be.

4 Application of Our Method

Step 1. Describe Scenario

For the elicitation of the context, we introduced so called *context-patterns* in earlier works of ours [21–24]. We also published the initial steps towards a pattern language for context-patterns [1]. We created a *Smart Home* context-pattern that is specifically based on a particular scenario NeSSoS industrial partners are considering.

[10] Note that a Prosumer is an energy consumer, who also sells small amounts of energy to the energy provider.

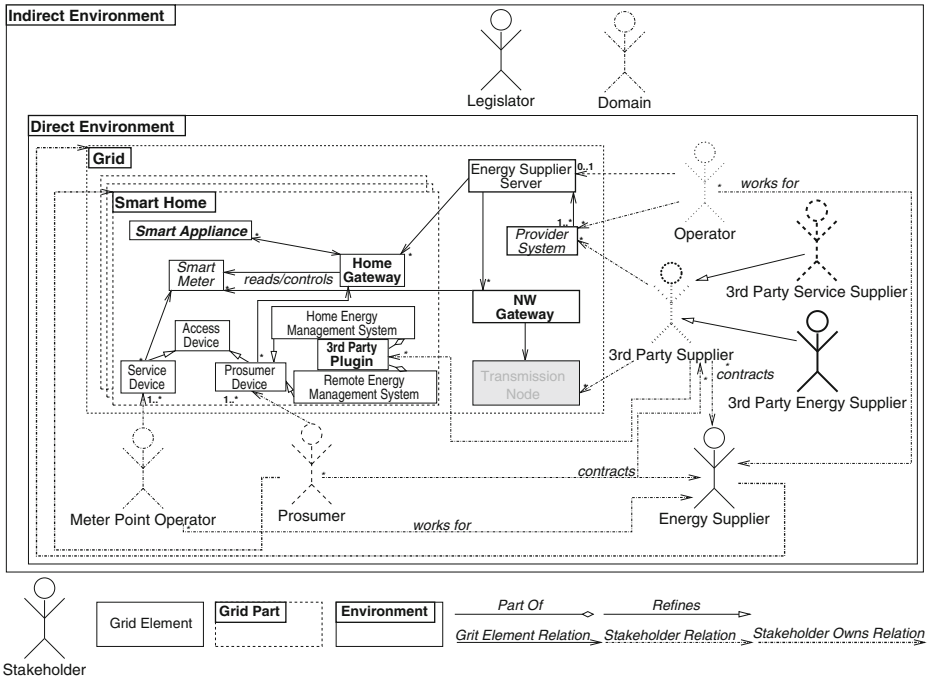


Fig. 2. Smart Home General Pattern

This context-pattern is a refinement of our general smart grid pattern, which was described based on a in deep analysis of several documents like the CC protection profiles for Smart Meters [25,26], the documentation of the OpenNode project [27,28], the documentation of the OpenMeter project [29], the industry case studies from the NESSoS project, and the Canadian smart grid implementation program [30,31]. The general pattern is available in one of our publications [1].

We depict our smart home context-pattern in Fig. 2. The pattern is divided into four major parts. The first part is the *grid* in which the smart home will be integrated. The *direct environment* contains all the *direct stakeholders*, who have a direct relation to one or more parts of the grid. Hence, they are able to directly influence the grid. In contrast, the *indirect stakeholders* of the *indirect environment* have no influence and, in most cases, also no interest in the grid parts and elements themselves. But they have an influence on the direct stakeholders, and therefore they are important for the system-to-be. The *Smart Home* contains the system to be built. This does not necessarily mean that all parts of the smart home are object of an development project, but at least one will be the machine to be built. The grid, the direct environment, the indirect environment, and the smart home are object to be described to get an understanding of the system-to-be and its context.

Note that all stakeholders are represented by stick-figures. To distinguish the different types of stakeholders in an smart home pattern later on, each type's stick-figure representation has its own line style. There are two kinds of important indirect stakeholders. First, the *domain* which represents further specific domains, beside the smart grid domain, for which the system-to-be is developed. The domains influence is based on self-regulations of a domain, standards for this domain and so forth. Second, the *legislator* describes the government of a country for example. A legislator enacts and enforces different regulations which the system-to-be has to be compliant to.

For the direct stakeholders there are five kinds of importance. The *Prosumer contracts* the *Energy Supplier* and/or the *3rd Party Energy Supplier* to buy energy and grid services. In addition, the *Prosumer* can also sell small amounts of energy to the *Energy Supplier* and/or the *3rd Party Energy Supplier*, which is a *3rd Party Supplier*. The amount of bought and sold energy is measured by the *Smart Meter*. The *Prosumer* reads the energy values using a *Prosumer Device*. Two special Prosumer devices are the *Home Energy Management System* and the *Remote Energy Management System*, which allow, besides the viewing of energy values, the configuration of *Smart Appliances*. *Smart Appliances* are configurable devices such as heaters, which can be configured to turn on at a specific time or when certain conditions arise, for example, a certain temperature. To extend the functionality of the remote/Home Energy Management Systems, the Prosumer can buy *3rd Party Plugins* from different *3rd Party Service Providers*. This can be simple GUi services for viewing information, but also complex new functionality, which e.g. requires a permanent internet access to get information from the environment like weather data. All the communication between the smart home elements is coordinated via the *Home Gateway*. One exception are *Service Devices* used by the *Meter Point Operator*.

The meter point operator *works for* the energy supplier. His/Her tasks are installing and maintaining the devices at the consumer side, in particular, the Smart Meter via service devices. They are a type of *Access Devices* like the Prosumer devices, but with special abilities. Access Devices are directly connected to the Smart Meter.

The *Operator* also *works for* the energy supplier and executes different tasks, e.g., maintenance or billing using legacy *Provider Systems* and the *Energy Supplier Server*. The provider system and the energy supplier server connect to the smart home using a dedicated channel provided by the *NW (Network) Gateway* or directly via the internet. The NW Gateway also communicates with a *Transmission Node*. We marked the transmission node in gray in this pattern, because we will not consider it for the remainder of this paper. The other technical grid elements are described in more detail in Table 7.

We illustrate our *scope template* in Table 1. The first column states the name of the stakeholders or grid elements, the next column states if the stakeholders or grid element is part of the scope, and the last column defines why a stakeholders or grid element is part of the scope or not.

Table 1. Scope template

Smart home pattern element	Part of scope	Reasoning
Stakeholder		
State the name of the stakeholder	Yes or No	Explain why the stakeholder is part of the scope or not
Grid elements		
State the name of the grid element	Yes or No	Explain why the grid element is part of the scope or not

Example Smart Home Scenario[11] **-** We illustrate a Smart Home scenario that industrial partners of the NeSSoS project are considering in Fig. 3. In our example, the threat analysis is conduced by an energy provider called *Tesla AG* and it is conducted on behalf of the *Tesla Prosumer*. Tesla wants to find out if the equipment and operations they apply to the Tesla Prosumer's Smart Home is secure to operate and does not harm the Tesla Prosumer's privacy concerns. Tesla excludes any equipment that they did not recommend or provide from the scope of the threat analysis. The elements in the scope are listed in Table 2.

This scenario considers the German Law as the binding law, because it concerns a release of a Smart Grid specifically tailored to German Prosumers. Hence, we instantiate the legislator *Germany* and since privacy concerns are relevant, we refer to the German Federal Data Protection Act (BDSG). Moreover, regulations for the *Energy* domain have to be obeyed, such as the electricity- and gas-supply act (Energiewirtschaftsgesetz, EnWG), as well as laws regarding the protection of the environment (*Nature Protection*), like the German Renewable Energy Act (Erneuerbare-Energien-Gesetz, EEG).

Tesla uses a *Tesla Server* for the electronic communication with the smart home and in particular the *Wan+WLan+Lan Router* hardware of the Tesla Prosumer. The Tesla server is maintained by the *Tesla Service Staff Member* and it is connected with the *Sunshine System*, the server of the *Sunshine Inc*, which is a subcontractor of the Tesla AG. In addition, the *Energy Meter* that is provided by Tesla also communicates directly with the *Tesla NW Gateway*. The Tesla Prosumer uses several Smart Appliances: A *Thermostat*, a *Smart TV*, and a *Solar Collector*. The Energy Meter is maintained by the *SmartSpecialist KG* using a *Meter Display & Interface* and a *Meter Calibration Tool*. The Tesla Prosumer uses a *Home Energy Management System* to control his/her Smart Appliances when he/she is at home. When the Tesla Prosumer is not at home he/she uses the *Remote Energy Management System* to control his/her Smart Appliances. Furthermore, the Tesla Prosumer uses a *Weather Controlled Heating Plugin* for the Home Energy Management System to automate the temperature regulation of the smart home. This plugin is provided by the *Smart Apps* company.

[11] All organizations appearing in this work are fictitious. Any resemblance to real organizations, companies or persons is purely coincidental.

Table 2. Scope template instance

Smart home pattern element	Part of scope	Reasoning
Stakeholder		
Smart Specialist KG	No	Only visits the smart home for maintenance
Tesla Prosumer	Yes	
Tesla AG	No	Does not reside in the smart home
Tesla Service Staff Member	No	Does not reside in the smart home
Sunshine Inc.	No	Does not reside in the smart home
Smart Apps	No	Does not reside in the smart home
Grid Elements		
Tesla Server	No	Is outside the smart home
Tesla NW Gateway	No	Is outside the smart home
Sunshine System	No	Is outside the smart home
Wan+Wlan+LAN+Router	Yes	The router is provided by the *Tesla AG*
Solar Collector	No	Not provided or recommended by *Tesla AG*
Smart TV	No	Not provided or recommended by *Tesla AG*
Thermostat	No	Not provided or recommended by *Tesla AG*
Energy Meter	Yes	Is inside the smart home and provided by *Tesla AG*
Meter Display and Interface	Yes	Is inside the smart home and provided by *Tesla AG*
Meter Calibration Tool	No	It is only inside the smart home when the *SmartSpecialistKG* conducts maintenance on the *Smart Meter*
Home Energy Management System	Yes	Is inside the smart home and provided by *Tesla AG*
Remote Energy Management System	Yes	Is inside the smart home and provided by *Tesla AG*
Weather Controlled Heating Plugin	No	Is inside the smart home, but not provided by *Tesla AG*

Step 2. Identify Assets

We identify the assets in our scope and use our asset template (see Table 3) to document them.

We show an instantiated asset template for the *Home Energy Management System* (see Table 4) and refer for the remaining instantiations to our technical report (see Footnote 5).

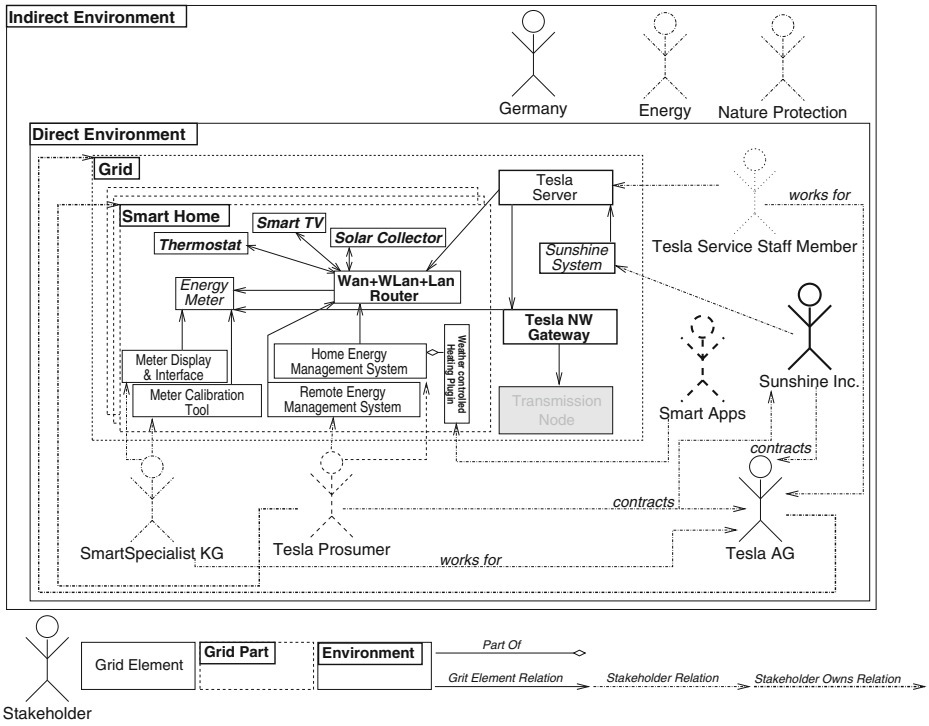


Fig. 3. Smart Home Instantiation Pattern

Step 3. Create DFDs

The DFD model helps to analyze the scenario and to identify crucial data flows for the definition of entry points, security requirements, possible threats (not only by external attackers but also by insiders or normal participants of the system). The Data Flow Diagram (DFD) depicts which information flows between which

Table 3. Asset template

Asset	State the name of the asset
Reasoning	Argue why this asset has a value for a stakeholder in the scope
External dependency	State the external system or stakeholder on which the asset depends
Security assumptions	State security assumptions about the environment of the asset
Security notes	State security notes for the environment of the asset
Contains assets	Are other assets part of this asset?

Table 4. Instantiated asset template for the Home Energy Management System

Asset	Home Energy Management System
Reasoning	The Home Energy Management Systems controls Smart Appliances and is the communication terminal for the Prosumer with *Tesla* and other energy providers
External dependency	The Home Energy Management Systems relies on the *Wan+Wlan+LAN+Router* to provide the communication infrastructure and to support proper confidentiality and authentication mechanisms
Security assumptions	The *Wan+Wlan+LAN+Router* is configured and maintained reliably
Security notes	The *Remote Energy Management System* and the *Wan+Wlan+LAN+Router* are configured to use proper confidentiality and authentication mechanisms
Contains assets	Remote Energy Management System

Table 5. Description of DFD elements according to [2]

DFD element type	Description
A double circle is a *Complex Process*	A representation of a process that performs different operations
A circle is a *Process*	A representation of a process that performs one discrete task
A rectangle is an *External Entity*	Something the SuA requires, but does not control
Parallel lines are a *Data Store*	Persistent data storage that the SuA uses
An arrowed line is a *Data Flow*	Means of data transmission throughout the SuA
A dotted line is a *Privilege Boundary*	Privilege Boundary represent data moving between different trust levels

interfaces. Figure 5 represents the information flow between the identified assets, including processes, storage, interfaces and elements of a smart home. Elements are depicted as described in Table 5.

For creating the DFD we use the smart home pattern instance (Fig. 3) as an input. The DFD is then created in two phases: the *mapping of the smart home pattern instance* to a generic DFD, and the *refinement of the generic DFD* with information about data storage and specific, additional processes. Note that the DFDs focus on technical elements, and thus, all stakeholders are left out, and the relations between stakeholders are not considered in DFDs. Hence, the resulting DFD will be a refinement of the smart home pattern instance showing only a technical point of view, adding the information about involved data and its flows.

Table 6. Mapping context-pattern elements to DFD elements

Smart Home Pattern Element	Part of Smart Home	Part of Scope	DFD Element
Stakeholder	☐	☐	Not Mapped
Grid Element	☑	☑	Complex Process
Grid Element	☑	☐	External Entity
Grid Element	☐	☐	External Entity
Grid Element Relation	☐	☑	Data flow
Grid Element Relation	☐	☐	Data flow
Stakeholder Relation	☐	☐	Not mapped

☑ : yes, ☐ : No, ☐ : not relevant

Mapping of the Smart Home Pattern Instance. For the general mapping of the smart home pattern instance to the generic DFD, we use a mapping table (see Table 6). We leave out all stakeholders and stakeholder relations for the aforementioned reasons. Thus, there is no mapping for them.

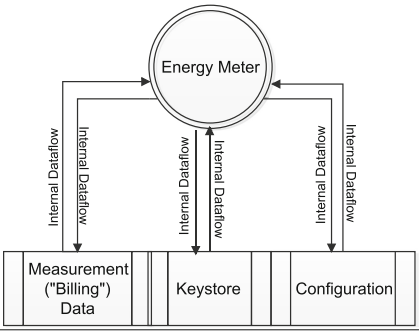
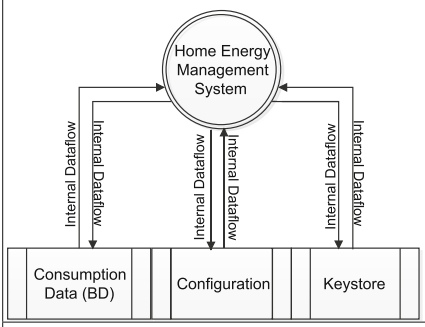
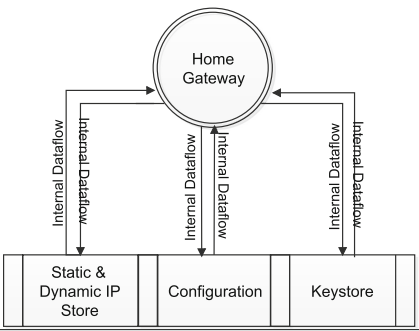
Note: Starting from this point, we exemplify our method on only the most important elements (see Fig. 4) of the scenario (see Fig. 3). A more complete mapping and analysis of the scenario is presented in the extended technical report[12].

As a **first step**, we focus on those grid elements of the DFD, which are clearly part of the scope. They are represented as complex processes. Table 7 contain suggestions on how to model an element of the smart home pattern instance, depending on which element of the smart home pattern it instantiates. For example, Table 7 suggests to add the Smart Meter instance as a complex process with three data stores (*Measurement (Billing) Data*, *Keystore*, and *Configuration*) and the corresponding data flows. Hence, the instantiated element "Energy Meter" of the smart home pattern instance (see Fig. 3), is represented in the DFD as a complex process called *Energy Meter* with the data stores *Energy Meter Keystore*, *Energy Meter Application Data*, *Energy Meter Measurement (Billing) Data* (see Fig. 5).

The **second step** is to consider elements inside the smart home, which are relevant in a security perspective, but cannot be actively changed, because they are provided by external third parties. They are modeled as external entities inside the smart home. All elements added in this step have to be separated from the elements added in Step 1 using privilege boundaries. The reason is, that the smart home has several stakeholders. We analyze the core components of the smart home, which are usually provided by one party and related subcontractors, such as the energy provided and meter point operators. The level of trust for parts that cannot be managed actively by those, is thus different to elements, that interact within the smart home, but are provided by external, heterogeneous parties. For example, the Smart TV, which is an element in the smart home (see the pattern instance in Fig. 3), is an element that interacts with other components, but is not part of the scope, as it is provided by an external

[12] The technical report can be found at: http://www.uml4pf.org/publications/smart-home.pdf.

Table 7. Suggestion for modeling elements in scope

Grid Element	
Suggested DFD Element(s)	**Reasoning**
Smart Meter	
	The Smart Meter (SM) and its databases. The Smart Meter stores the energy consumption of appliances within a home environment individually. The Smart Meter stores the per appliance measurement as well as static cryptographic keys and certificates, to sign its messages as well as to secure the communication channels. The *Meter Display & Interface* (compare Figure 3) is contained in the Energy Meter complex process.
The Home Energy Management System (HEMS or EMS)	
	The Home Energy Management System (HEMS or EMS) and its databases. The EMS is the main controlling entity in the Smart Home. It visualizes how much the user's appliances or rooms are consuming, and controls the energy production and storage. Demand Side Management events [32] are also handled here. The required information is acquired from Smart Appliances and the Smart Meter, and is stored in one of the EMS databases. Key exchange and deploying management for new and existing Smart Appliances is handled by the EMS as well.
Home Gateway	
	Home Gateway and its databases. HGs are devices that can access the Internet, and also via the Home Area Network, the Smart Appliances, electric switches, and the Smart Meter. They connect every entity in the Smart Home and are responsible for routing messages from one entity to another. HGs store, apart from their application configuration, cryptographic keys for the Home Area Network communication and the external forwarding of Billing Data Feedback (BDF) requests (generated by the EMS) to the Energy Supplier.

(non-trusted) manufacturer. Hence, it is added as external entity separated by a privilege boundary (see Fig. 5).

The **third step** is to add the grid elements, which external to the smart home, but are still relevant in a security point of view. Note that we introduce the

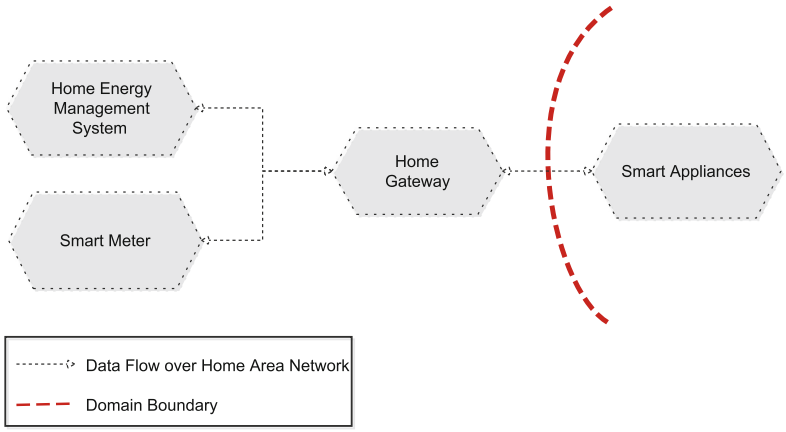


Fig. 4. Generic DFD for Smart Home

generic DFD with generic placeholders for the elements inside the smart home (see Fig. 4), and do not substitute them yet by the instatiations suggested in Table 7. However, the elements inside the smart home, that are provided by third parties, are shown as external entities and will remain so throughout the analysis. In a latter step, all this elements will be substituted by their instantiations. For example the *Home Gateway* in Fig. 4 is replaced with the *WLAN, LAN, WAN Router* from Fig. 3.

The **fourth step** is to add the grid element relations contained in the smart grid pattern instance to the DFD and the grid elements, which are not part of the smart home. Basically, each grid element relation, which is part of the scope, is mapped to at least one data flow. A grid element relation is part of the scope, if at least one of the connected grid elements is part of the scope. It is mapped to one data flow, if it is unidirectional. Otherwise, it is mapped to two data flows. Figure 4[13] comprises all elements in their first, generic representation and their mapped relations (Table 9).

Refinement of the Initial DFD. The initial DFD as modeled in Step 1 to 4 can be refined further where ever needed. Data stores can be split up to refine assets, or central processes are added. For example, we added the process Internet Routing to the DFD shown in Fig. 5 (see Footnote 13). From the intersection of elements in Figs. 3 and 5, a list of refined assets can be derived. One refined asset is detailed in Table 8. The representation of the refined assets corresponds to the asset template presented in Step 4. The Prosumer interaction with the EMS in his/her premises. The data flow diagram already captures some aspects of security, which helps to further identify possible assets. Every component has a cryptographic keystore, which stores any cryptographic information needed

[13] Note that we simplified the model for readability purposes. The interested reader can find the complete model in our technical report (see Footnote 5).

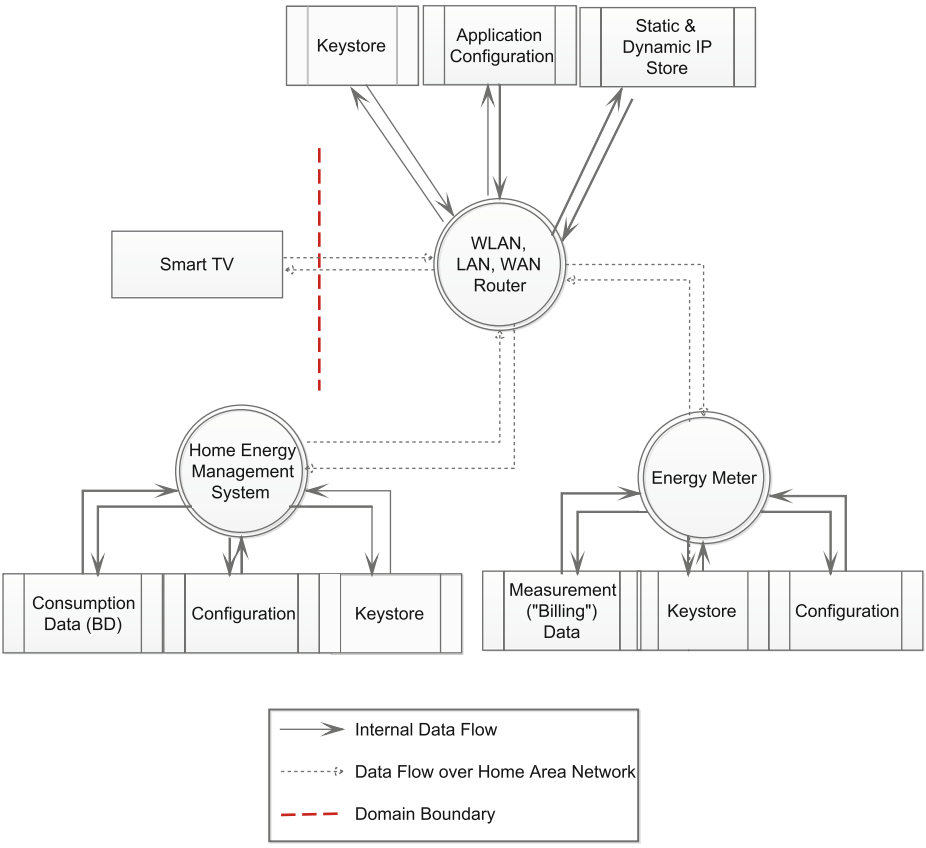


Fig. 5. DFD for a Smart Home Scenario

for signing messages and securing communication channels. Personally identifiable information such as Billing Data and customer profile data is also depicted (see Fig. 5). It should be noted that underlying protocols such as key exchange, pairing and other protocols are not further discussed in this paper (Table 10).

Step 4. Identify Entry Points and Analyze Vulnerabilities

From the perspective of an attacker, the assets identified in the previous steps represent valuable targets. With all assets in mind, different *entry points* can be identified. Entry points describe a certain vulnerability, which can be exploited, creating an *attack tree* from the entry point to one or several assets. Figure 6 gives insight into the different entry points. It should be noted that entry points are elicited considering the security assumptions of each individual asset defined in the refined asset descriptions (e.g., see Table 8, "Security Assumptions").

Whether an element is possibly an entry point or not depends highly on attackers, their different motives and expertise. Different attacker classifications

Table 8. Asset: Home Energy Management System

Asset	Home Energy Management System
Reasoning	The Home Energy Management System controls Smart Appliances, processes & visualizes real-time Billing Data, reacts to Demand Side Management events and is the communication terminal for the prosumer with the Energy Supplier and other third parties
External dependency	The Home Energy Management System relies on the *Home Gateway* to provide the communication infrastructure and to support proper confidentiality and authentication mechanisms. Additionally, it has to rely on the Smart Meter's Billing Data and its correct energy measurement for energy management
Security assumptions	The *Home Gateway* and the *Smart Meter* are configured and maintained reliably. The Energy Management System does only allow the prosumer to interact with the user interface for energy management and does not allow to access or alter any other functionality. The EMS does only allow third party plugins to execute sandboxed algorithms, Demand Side Management does not allow direct load control (DLC)
Security notes	The Energy Management System should not be physically accessible by the prosumer. Solely the user interface (e.g., a touch screen) should be available
Contains assets	Cryptographic keys for authentication and communication with third parties, Billing Data in real-time frequency

Table 9. Asset: Smart Meter

Asset	Smart Meter
Reasoning	The Smart Meter's measurement affects the billing, energy management of the Prosumer, energy forecasts for individual sectors and value added services from third parties
External dependency	The *Smart Meter* relies partially on the *Home Gateway* for transmitting Billing Data to the Energy Management System
Security assumptions	The *Home Gateway* and the provides a stable connection and is a trusted device
Security notes	The *MPO* does not obtain any energy consumption data of the prosumer. The Smart Meter does not allow any interaction with the prosumer, Billing Data is acquired by means of the Energy Management System. The Smart Meter does not allow remote energy shutdown
Contains assets	Billing Data, cryptographic keys for message verification and for communicating with other parties

Table 10. Asset: Home Gateway

Asset	Home Gateway
Reasoning	The communication internally in the Smart Home and externally with the grid is based on the *Home Gateway*. Without the HG, the *Home Energy Management System* could neither receive Billing Data from the Smart Meter, nor manage Smart Appliance's behavior, nor send and receive Billing Data Feedback as well as react to Demand Side Management events
External dependency	The Home Gateway has to be available and configured properly by the supplier
Security assumptions	Proper configuration means that end point IP addresses are correct, that authentication is enforced and confidentiality of the data transmissions is adequate
Security notes	The Prosumer has to prevent that confidentiality of data transmissions are adequate and that authentication mechanisms are activated. Misbehavior needs to be notified to the MPO
Contains assets	Communication keys for the Home Area Network

Table 11. Assets: cryptographic keystores

Asset	Cryptographic Keystores
Reasoning	Cryptographic keystores were referenced in the sub assets section ("contains assets") of every asset described above. Cryptographic information assure message integrity, as well as confidentiality for the communication partners
External dependency	Cryptographic information depends on the underlying protocols for secret generation, key exchange and management
Security assumptions	Key storage is only accessible by internal data flows
Security notes	Billing Data and profile data should not be used for purposes other than contractual purposes
Contains assets	-

can be used in this step, e.g., classification by motivation as in [3]. An exemplifying set of expertise attackers is chosen here, namely the *network* and the *software attackers*, inside and outside the smart home. An exhaustive analysis of all attacker models, including *physical* and *social engineering* adversaries, will be considered in future work. **Network attackers** are adversaries that have access to a target network and can eavesdrop and modify its messages actively. They have limited computational capabilities, time as well as financial resources. They can be both, an authorized user or an external adversary. It is assumed that they cannot break any cryptographic challenges, nor are they

Table 12. Assets: personally identifiable information: customer profile data, billing data

Asset	Profile Data, Billing Data
Reasoning	Personally Identifiable Information (PII) like profile data (name, address, birthday, etc.) and Billing Data allow deep insight into the habits and affections of the PII's subject
External dependency	Billing Data depends on the Smart Meter measurement accuracy. Aggregated Billing Data depends on the aggregation process
Security assumptions	Smart Meter measurements are accurate. Aggregation algorithms are secure
Security notes	The cryptographic keystore is physically secured
Contains assets	-

able to penetrate physical locks nor break software security measures. **Software attackers** on the other hand, are able to analyze, reverse engineer and compromise software systems. They are not capable of interfering in network traffic, nor are they able to penetrate physical security. They have limited computational capabilities, time as well as financial resources and can be both, an authorized member of the system or an external adversary (Table 11).

For eliciting the possible entry points, we apply for each complex process a high level reasoning, if the aforementioned attacker types can access this particular process or not. If we cannot reject the assumption that any attacker can access the process at hand, it is marked as a general entry point. Next, we conduct for each process which is marked as general entry point, an entry point refinement. We check for each data flow from or to this process whether one of the possible attackers can potentially access it or not. If at least one attacker has access to the data flow at hand, we mark this data flow as an entry point. The result is shown in Fig. 6 (see Footnote 13). Warning triangles visualize each entry point in the Smart Home. An attacker will chose individual entry points depending on the asset(s) that he/she wants to compromise (Table 12).

With the elicitation of assets and entry points, vulnerabilities and possible threats can be derived. This is done in the following step by mapping entry points to assets and categorizing them according to the STRIDE taxonomy.

STRIDE stands for the following actions conducted by an attacker: *Spoofing*, e.g., the identity of a stakeholder; *Tampering* with data or code; *Repudiation* means plausible deniability of having performed an action; *Information disclosure* of access restricted data; *Denial of service* attacks; *Elevation of privilege* means an attacker gains an increased capability and gains admin (or root) capability. In Sect. 2.2, we introduced the identification of threats by mapping STRIDE threats to DFD elements. We use a vulnerability template (see Table 14) to describe the possible vulnerabilities associated with our entry points.

Table 13. Entry point elicitation table

Process	Possible Attackers	Reasoning
Wan+Wlan+LAN+Router	☑ Network Attacker ☐ Software Attacker	The router is connected to all kinds of networks of the smart home. Hence, the it is vulnerable to attacks against these networks. The software running on router cannot be changed or influenced without direct access.
Energy Meter	☑ Network Attacker ☐ Software Attacker	The Energy Meter is connected to the smart home using WLAN. Thus, it is accessible by an network attacker. The software cannot be changed and is tamper proofed.
Home Energy Management System	☑ Network attacker ☑ Software attacker	Is connected to the WLAN. Hence, accessible by the network attacker. And it is highly configurable and extensible with own code. Hence, it is prone to software attacks.

☑ : yes, ☐ : No

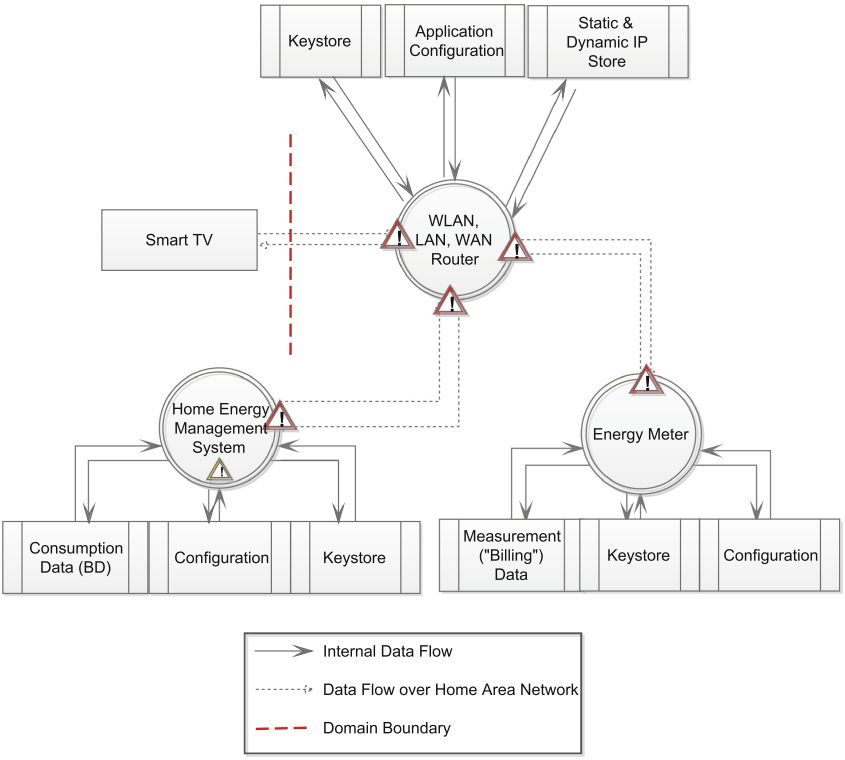


Fig. 6. Attacker Entry Points for the Smart Home

Table 14. Vulnerability template

Entry point	STRIDE threat	Attacker type	Reasoning
State the concerned entry point including relevant data flows and/or processes. For network attacker, data flows are always relevant and processes are optional, for a software attacker it's vice versa. The reason is that a network attacker considers the data flow first and afterwards can manipulate or use flows to manipulate a process. We add the process, if it is essential for this entry point to exist. For example, if a device provides root access rights to all incoming network connections it is essential for this entry point. In contrast, software attackers focus on exploits for source code, but may require data flows to, e.g., facilitate a data leak	State the considered STRIDE threat	State the concerned attacker type	Describe the threat instance

We illustrate one instantiated vulnerability templates for the *Home Energy Management System* (see Table 15). We refer for the remaining instantiations of our vulnerability template to our technical report (see Footnote 5).

Step 5. Determine Threats

In this final step of our threat analysis, we analyze how an attacker can possibly harm assets by using the entry points and their STRIDE threats elicited previously. For each asset we model at least one *Attack Path DFD*, which is a DFD that contains at least one threat caused by an attacker. All identified threats have to appear in at least one *Attack Path DFD*. The assets concerned in an *Attack Path DFD* are marked with a star symbol. The threat is modeled as a complex process that is marked in red and with the attacker symbol. This complex process exploits the entry points. We model these exploits using dotted lines

Table 15. Vulnerability template instance Home Energy Management System

Entry Point	STRIDE threat	Attacker type	Reasoning
Home Energy Management System (Process)	Spoofing	Software attacker	In a special scenario, Status & Control messages could be used to exploit the ***EMS***. The attacker could analyze over a large period of time every message in the *HAN* and learn possible new ways to *spoof* other elements of the Home Area Network, e.g., the Smart Meter. This could lead to information disclosure and denial of service
Home Energy Management System (Process)	Tampering	Software attacker	An attacker can manipulate user policies, Status & Command messages and change the behavior of Smart Appliances at his will. In a worst case scenario, the attacker could physically harm a person inside the home
Home Energy Management System (Process)	Repudiation	Software attacker	An attacker can override non-repudiation mechanisms to gain advantage of e.g. third party services
Home Energy Management System (Process)	Information disclosure	Software attacker	An attacker has access to the EMS' databases. This enables the disclosure of all HAN traffic and Billing Data generated in real-time
Home Energy Management System (Process)	Denial of service	Software attacker	An attacker is able to deny any communication with the EMS, sabotaging Demand-Side-Management events, control over Smart Appliances, and the Prosumer's energy management
Home Energy Management System (Process)	Elevation of privileges	Software attacker	The EMS supports *third party plugins*, which are allowed a sandboxed space in the EMS' functionality. If a malicious plugin is able to find a backdoor to the full EMS functionality, several assets could be compromised: *Billing Data and customer profile data* that identify the customer, *cryptographic keys* which allow proper authentication against the *Energy Supplier*, other *third parties* and the *Smart Meter*. The *EMS* controls the physical behavior of *Smart Appliances* which might endanger the appliance itself or the well being of persons inside the house

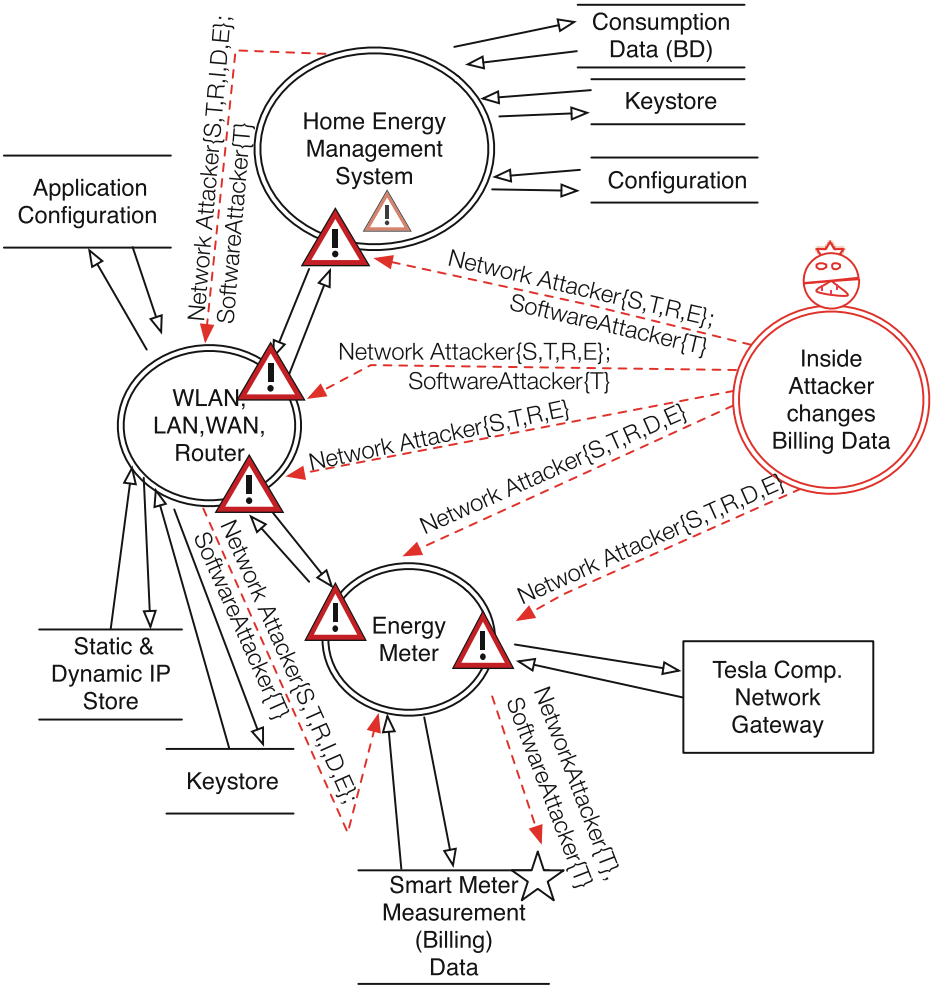


Fig. 7. Attack Path DFD: Inside attacker changes billing data

with filled arrows. The arrows are labeled with the attacker type followed by the exploited threat in curly brackets. If several attacker types have to be annotated on one exploit arrow, they are separated with a semicolon. In addition, we add exploit arrows at the processes containing the entry points to illustrate different paths towards the asset. Hence, the Attack Path DFD diagrams show multiple ways of how an attacker can harm an asset.

We present examples of an inside attacker that aims to change the billing data in Fig. 7 and an outside attacker in Fig. 8. An inside attacker in the smart home scenario is using only scope elements as entry points. In our example, an inside attacker could be an employee of Tesla or a resident of the smart home. Outside attackers are all other kinds of attackers.

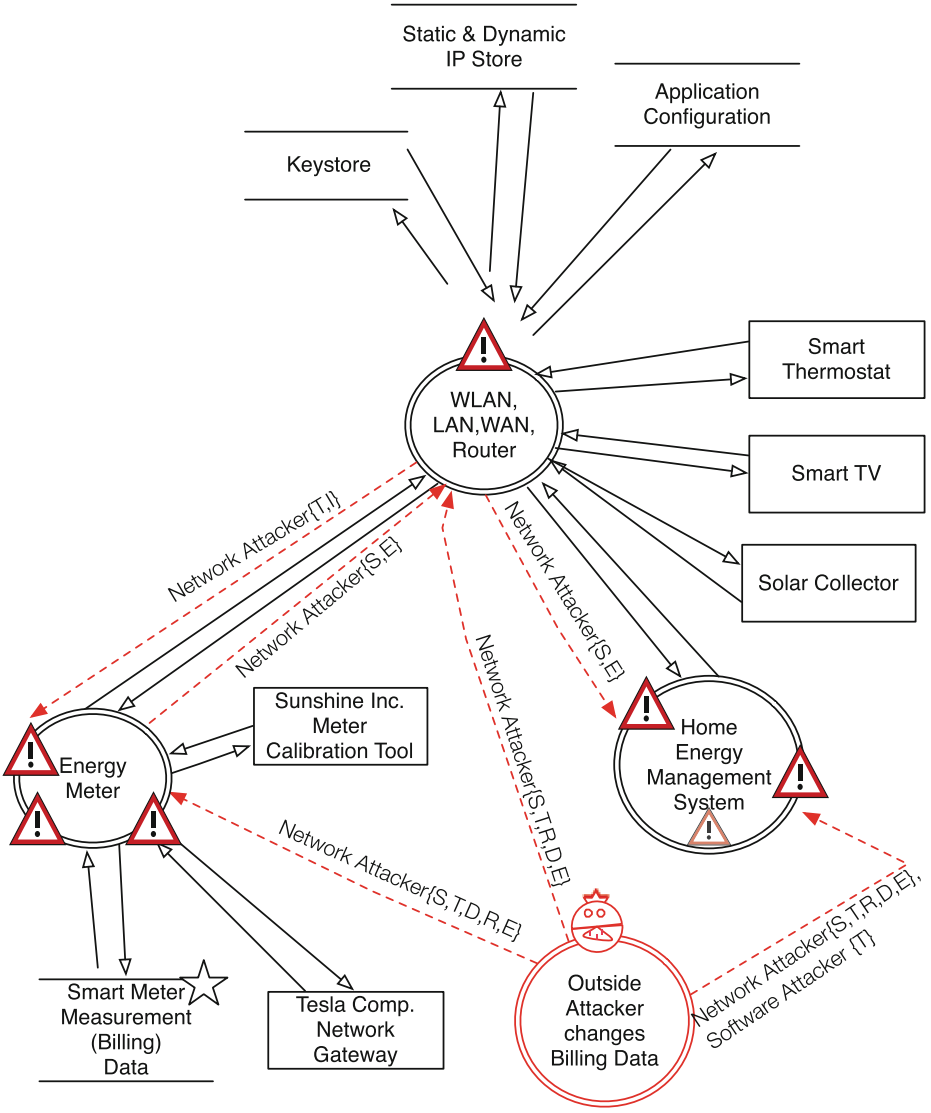


Fig. 8. Attack Path DFD: Outside attacker changes Billing Data

The attack path DFDs have to be analyzed for all possible attack paths. For example, the inside attacker with the goal to change billing data in the smart home (see Fig. 7) (see Footnote 13) can initiate spoofing by a network attacker at the entry point at the *Home Energy Management System* and pretend to be the *Energy Meter* that sends *Smart Meter Measurements*. Note that we did not show any STRIDE attacks that are not relevant for the attack process, such as information disclosure of *Smart Meter Measurements*. Tampering by a software

attacker with the *Smart Meter Measurements* results in an exploit of the *Home Energy Management System*, such as cross side scripting (XSS). The attack executes malicious code in the *Home Energy Management System*. This causes the *Home Energy Management System* to display wrong *Smart Meter Measurements (Billing) Data*, which might lead to the false demand side management events, causing grid instability or enabling economic advantages for an attacker. In order to prevent the *Energy Meter* to send an update on *Smart Meter Measurements* the attacker could also initiate a denial of service attack to the meter. Another example is that the inside attacker could spoof the *Energy Meter* by pretending to be the *Tesla Comp. Network Gateway*. In this case the inside attacker would be an employee of Tesla, who has access to the keys for the encrypted communication between the *Tesla Comp. Network Gateway* and the *Energy Meter*. The employee could order the *Energy Meter* to reset the *Smart Meter Measurements (Billing) Data* and cause a loss of information. Another possibility would be that the attacker changes the measurements during the transmission to the *Tesla Comp. Network Gateway*. We propose to compile a list of attack paths that uses every entry point at least once.

We also provide examples concerning an outside attacker (see Fig. 8) (see Footnote 13). An outside attacker could use the *Internet Routing* to connect to the *WLAN,LAN,WAN, Router* process, e.g., via Spoofing as the *Tesla Comp. Server*. From there an attacker could try to move to the *Energy Meter* and pretend to be the *Tesla Comp. Server*. This could lead to a reset of the routing information, e.g., the IP-Address of the Tesla Comp. Network Gateway could be changed via a specific command. Normally this should only be possible via the *Sunshine Inc. Meter Calibration Tool*, however the attacker can conduct a denial of service attack on the flow between the Meter and the *Tesla Comp. Network Gateway*. When the Meter cannot contact the Tesla Comp. using the *Tesla Comp. Network Gateway* for more than 24 h, the *Energy Meter* accepts connections from the *Tesla Comp. Server* via the *WLAN,LAN,WAN, Router* with equal privileges.

5 Conclusion

We contribute a method for threat analysis of smart home scenarios. Our work is based on the threat analysis of the Microsoft Security Development Lifecycle (SDL), which is becoming a recognized best practice methodology. In particular, we provide patterns and templates that help to elicit and analyze domain knowledge and can be re-used for different projects. We illustrated our method on a smart home scenario that the industrial partners of the NESSoS project are considering.

The main benefits of our methods are as follows:

- A structured threat analysis method that refines the approach of the Microsoft SDL.
- Smart Home pattern for structured domain knowledge elicitation of different smart home scenarios.

- Scope and asset templates that refer to the elements of the smart home pattern and contain the demanded descriptions of the Microsoft SDL of these elements.
- A DFD pattern derived from the smart home pattern that can also be instantiated for different scenarios.
- Templates to describe assets and entry points into the system.
- Attack Path DFDs that illustrate how an attacker can move in the system from entry points to an asset in order to harm it.

In the future, we will formalize the threat analysis to enable computer-aided support for our threat analysis. Moreover, we want to conduct a controlled experiment with practitioners. Some of them shall use our method and some will apply the Microsoft SDL without our support. We will compare the results to figure out if our method reduces the workload of software engineers and at the same time enhances and/or refines threat findings.

References

1. Beckers, K., Faßbender, S., Heisel, M.: A meta-model approach to the fundamentals for a pattern language for context elicitation. In: Proceedings of the 18th European Conference on Pattern Languages of Programs (Europlop), ACM (2013) (Accepted for Publication)
2. Howard, M., Lipner, S.: The Security Development Lifecycle: SDL: A Process for Developing Demonstrably More Secure Software. Microsoft Press, Cambridge (2006)
3. Aloula, F., Al-Alia, A.R., Al-Dalkya, R., Al-Mardinia, M., El-Hajj, W.: Smart grid security: threats, vulnerabilities and solutions. Int. J. Smart Grid Clean Energy **1**(1), 1–6 (2012)
4. Lin, H., Fang, Y.: Privacy-aware profiling and statistical data extraction for smart sustainable energy systems. IEEE Trans. Smart Grid **4**(1), 332–340 (2013)
5. NIST: Guidelines for smart grid cyber security (2010)
6. Geer, D.: Are companies actually using secure development life cycles? Computer **43**(6), 12–16 (2010)
7. Win, B.D., Scandariato, R., Buyens, K., Grégoire, J., Joosen, W.: On the secure software development process: Clasp, {SDL} and touchpoints compared. Inf. Softw. Technol. **51**(7), 1152–1171 (2009). Special Section: Software Engineering for Secure Systems Software Engineering for Secure Systems
8. SANS: Sans - a member of the microsoft security development lifecycle (sdl) pro network (2014). http://www.sans.org/security-resources/microsoft-sdl
9. OWASP: CLASP (Comprehensive, Lightweight Application Security Process). Technical report, The Open Web Application Security Project (OWASP) (2011). https://www.owasp.org/index.php/Category:OWASP_CLASP_Project
10. Commission of the European communities.: Communication from the commission to the european parliament, the council, the European economic and social committee and the committee of the regions (2011)
11. Lu, Z., Lu, X., Wang, W., Wang, C.: Review and evaluation of security threats on the communication networks in the smart grid. In: Military Communications Conference, 2010 - MILCOM 2010, pp. 1830–1835 (2010)
12. Wang, W., Lu, Z.: Survey cyber security in the smart grid: survey and challenges. Comput. Netw. **57**(5), 1344–1371 (2013)

13. Yang, Y., Littler, T., Sezer, S., McLaughlin, K., Wang, H.: Impact of cyber-security issues on smart grid. In: 2011 2nd IEEE PES International Conference and Exhibition on Innovative Smart Grid Technologies (ISGT Europe), pp. 1–7 (2011)
14. McDaniel, P., McLaughlin, S.: Security and privacy challenges in the smart grid. IEEE Secur. Priv. **7**(3), 75–77 (2009)
15. Tøndel, I.A., Jaatun, M.G., Line, M.B.: Security threats in demo steinkjer - report from the telenor-sintef collaboration project on smart grids. Technical report, SINTEF/NTNU (2012)
16. Dhillon, D.: Developer-driven threat modeling: lessons learned in the trenches. IEEE Secur. Priv. **9**(4), 41–47 (2011)
17. ISO/IEC: Information technology - Security techniques - Information security management systems - Requirements. ISO/IEC 27001, International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC), Geneva, Switzerland (2005)
18. Swiderski, F., Snyder, W.: Threat Modeling. Microsoft Press, Redmond (2004)
19. Beckers, K., Côté, I., Hatebur, D., Faßbender, S., Heisel, M.: Common criteria compliAnt software development (CC-CASD). In: Proceedings 28th Symposium on Applied Computing, pp. 937–943. ACM (2013)
20. Beckers, K., Hatebur, D., Heisel, M.: A problem-based threat analysis in compliance with common criteria. In: Proceedings of the International Conference on Availability, Reliability and Security (ARES), pp. 111–120. IEEE Computer Society (2013)
21. Beckers, K., Küster, J.C., Faßbender, S., Schmidt, H.: Pattern-based support for context establishment and asset identification of the ISO 27000 in the field of cloud computing. In: Proceedings of the International Conference on Availability, Reliability and Security (ARES), pp. 327–333. IEEE Computer Society (2011)
22. Beckers, K., Faßbender, S.: Peer-to-peer driven software engineering considering security, reliability, and performance. In: Proceedings of the International Conference on Availability, Reliability and Security (ARES) - 2nd International Workshop on Resilience and IT-Risk in Social Infrastructures(RISI 2012), pp. 485–494. IEEE Computer Society (2012)
23. Beckers, K., Faßbender, S., Heisel, M., Meis, R.: Pattern-based context establishment for service-oriented architectures. In: Heisel, M. (ed.) Software Service and Application Engineering. LNCS, vol. 7365, pp. 81–101. Springer, Heidelberg (2012)
24. Beckers, K., Faßbender, S., Küster, J.-C., Schmidt, H.: A pattern-based method for identifying and analyzing laws. In: Regnell, B., Damian, D. (eds.) REFSQ 2011. LNCS, vol. 7195, pp. 256–262. Springer, Heidelberg (2012)
25. BSI: Protection Profile for the Gateway of a Smart Metering System (Gateway PP). Version 01.01.01(final draft), Bundesamt für Sicherheit in der Informationstechnik (BSI) - Federal Office for Information Security Germany, Bonn, Germany (2011). https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/SmartMeter/PP-SmartMeter.pdf?_blob=publicationFile
26. BSI: Protection Profile for the Security Module of a Smart Meter Gateway (Security Module PP). Version 1.0), Bundesamt für Sicherheit in der Informationstechnik (BSI) - Federal Office for Information Security Germany, Bonn, Germany (2013). https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/SmartMeter/PP_Security_%20Module.pdf?_blob=publicationFile
27. OPEN node project: Evaluation of general requirements according state of the art. Technical report, OPEN node project (2010)
28. OPEN node project: Functional Use cases. Technical report, OPEN node project (2011)

29. OPEN meter project: D1.1 Requirements of AMI. Technical report, OPEN meter project (2009)
30. Department of Energy and Climate Change: Smart metering implementation programme, response to prospectus consultation, overview document. Technical report, Office of Gas and Electricity Markets (2011)
31. Department of Energy and Climate Change: Smart metering implementation programme, response to prospectus consultation, design requirements. Technical report, Office of Gas and Electricity Markets (2011)
32. Mohsenian-Rad, A.H., Wong, V., Jatskevich, J., Schober, R., Leon-Garcia, A.: Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid. IEEE Trans. Smart Grid **1**(3), 320–331 (2010)