Exploration of the Potential of Process Mining for Intrusion Detection in Smart Metering

Günther Eibl[1], Cornelia Ferner[1], Tobias Hildebrandt[2], Florian Stertz[2],
Sebastian Burkhart[1], Stefanie Rinderle-Ma[2] and Dominik Engel[1]
[1]*Josef Ressel Center for User-Centric Smart Grid Privacy, Security and Control, Salzburg University of Applied Sciences, Urstein Süd 1, A-5412 Puch/Salzburg, Austria*
[2]*Research Group Workflow Systems and Technology, University of Vienna, Währingerstrasse 29, A-1090 Vienna, Austria*
firstname.lastname@en-trust.at[1], *firstname.lastname@univie.ac.at*[2]

Keywords: Process Mining, Intrusion Detection, Smart Grids, Smart Metering.

Abstract: Process mining is a set of data mining techniques that learn and analyze processes based on event logs. While process mining has recently been proposed for intrusion detection in business processes, it has never been applied to smart metering processes. The goal of this paper is to explore the potential of process mining for the detection of intrusions into smart metering systems. As a case study the remote shutdown process has been modeled and a threat analysis was conducted leading to an extensive attack tree. It is shown that currently proposed process mining techniques based on conformance checking do not suffice to find all attacks of the attack tree; an inclusion of additional perspectives is necessary. Consequences for the design of a realistic testing environment based on simulations are discussed.

1 INTRODUCTION

In an effort to modernize traditional energy grids and move towards smart grids, information and communication technologies are used to enable communication and real-time interaction between producers, consumers and other stakeholders. In this transition smart meters play an important role, as they enable the inclusion of end-user homes.

While smart grids have the potential to achieve substantial benefits, also new threats arise (Berthier et al., 2010). In particular, the new possibility for a remote turnoff can also be seen as a vulnerability that could be exploited by an attacker to accomplish a large scale turnoff. Especially for critical infrastructures, there is a need to investigate possible weaknesses and attacks before being put into operation. This implies that envisaged processes should be modeled and checked for weaknesses. This paper explores the potential of process mining for intrusion detection in smart metering with particular emphasis on detecting intrusions that may lead to a large-scale turnoff.

Process mining with subsequent conformance checking for anomaly detection has been suggested for security enhancement (Van der Aalst and de Medeiros, 2005). However, anomaly detection is not specific enough. Anomalies are created using simple heuristics such as by interchanging of events (Bezerra et al., 2009), (Bezerra and Wainer, 2013) which is likely not describing anomalies arising from intrusions. Additionally, since conformance checking only considers the control flow the benefit of this technique is still unclear. Anomaly detection using process mining was combined with misuse detection considering four types of attacks related to the organizational perspective (Jalali and Baraani, 2012). Conformance checking can be applied for the detection of violations of security requirements like e.g. Separation or Binding of Duties requirements (Accorsi and Stocker, 2012).

This paper has two main contributions: instead of only creating anomalies, attacks on the smart meter shutdown process are more realistically and systematically derived and consequences for future enhancement of the evaluation methodology are discussed. Shortcomings of the current process mining methodology using conformance checking for intrusion are shown yielding consequences for future enhancement of the intrusion detection methodology.

The paper is structured as follows. Background about process mining, attack-defense trees and evaluation by simulations are provided in Section 2. In Section 3 the shutdown use case is modeled, attacks are systematically derived and detection methods based

on process mining are proposed. Several general findings, benefits and limitations of the method are discussed in Section 4. Finally, Section 5 contains conclusion and outlook.

2 PRELIMINARIES

2.1 Process Mining

The goal of process discovery is to automatically discover a process model by analyzing logs of recorded process events (Van der Aalst, 2011). The process model that is found can then be analyzed in order to improve or correct the process.

A *case* (or *instance*) is a specific execution of a process. Each case basically consists of a sequence of *activities* (or *events*) which are typically identified by their name, e.g. "Breaker interrupts power supply" (Figure 1). The process model describes the decisions that influence the process path and the ordering of activities (control-flow perspective).

Process instances can be described by additional attributes. The organizational perspective focuses on users and their roles and how they influence the process, e.g., it specifies which person performed the activity. The case perspective looks at properties or data of cases like a customer's energy consumption over a certain time interval. The time perspective describes, e.g., durations of activities and is particularly suited to detect bottlenecks.

In *conformance checking*, executed process instances are compared to a process model. A case is considered valid, if it can be created by the process. By trying to replay the case it is checked, if the ordering of its activities is compatible with the process. Conformance checking can be used in two different ways: on one hand it can be used to rate the *process* by the proportion of valid traces of the event log. On the other hand, for anomaly detection a *trace* is considered an anomaly if it is not a valid instance of the process.

2.2 Attack-Defense Trees

An attack tree (Salter et al., 1998) is an intuitive visual representation of attacks combined with formal semantics and algorithms that enable qualitative and quantitative analysis. The root of an attack tree is the goal of the attack. The tree is then iteratively constructed by dividing each goal into sub-goals where either all sub-goals must be reached to reach the goal (AND) or any sub-goal suffices (OR). The refinement is repeated until all sub-goals can be reached by basic actions. Attack trees have been extended in many different directions (Kordy et al., 2014).

Since in this paper not only attack but also detection methods should be modeled, an attack tree is not sufficient as a modeling structure. Based on the survey (Kordy et al., 2014), two different approaches can be identified as prime candidates for modeling both the attack and the detection method. An attack countermeasure tree (ACT) (Roy et al., 2012) can model attack, detection and mitigation events at any part of the tree and calculate probabilities of events. Attack defense trees (ADT) (Kordy et al., 2012) can model interleaving attacker and defender actions and provides several general methods for attribute computation like *probability of success*. While attack defense trees have the advantage that defensive actions can be refined, attack countermeasure trees have the advantage that both detection and mitigation events can be modeled.

2.3 Evaluation of Process Mining-Based Intrusion Detection by Simulations

Similar to existing approaches we propose to study and evaluate intrusion detection using simulations. There are several reasons for using simulations: (i) in contrast to a real system ground truth is available which permits a comparison of different detection methods, (ii) an assessment is possible prior to a roll-out which is important especially for highly critical infrastructures and (iii) variants of the system and its processes can be compared beforehand.

A simulation environment for intrusion detection needs to simulate (i) the process with its variations and (ii) effects of intrusions. For both *normal* and *anomalous* process instances, execution logs are being created. While the former can be achieved in a typical process mining environment, the latter needs an additional effort.

As a first step in that direction, both the model and the resulting traces can be transformed (Stocker and Accorsi, 2013). Model transformations persistently change the model by transforming AND-gateways to XOR-gateways and vice versa or swapping of two activities. Trace transformations are then performed on valid traces of the transformed model. Some trace transformers work by simply changing delays in the process execution or skipping of activities. Other trace transformations create policy violations related to the organizational perspective like e.g. authentication or binding-of-duties violations.

In this paper, the effect of intrusions aiming at a smart meter shutdown on traces of the smart meter

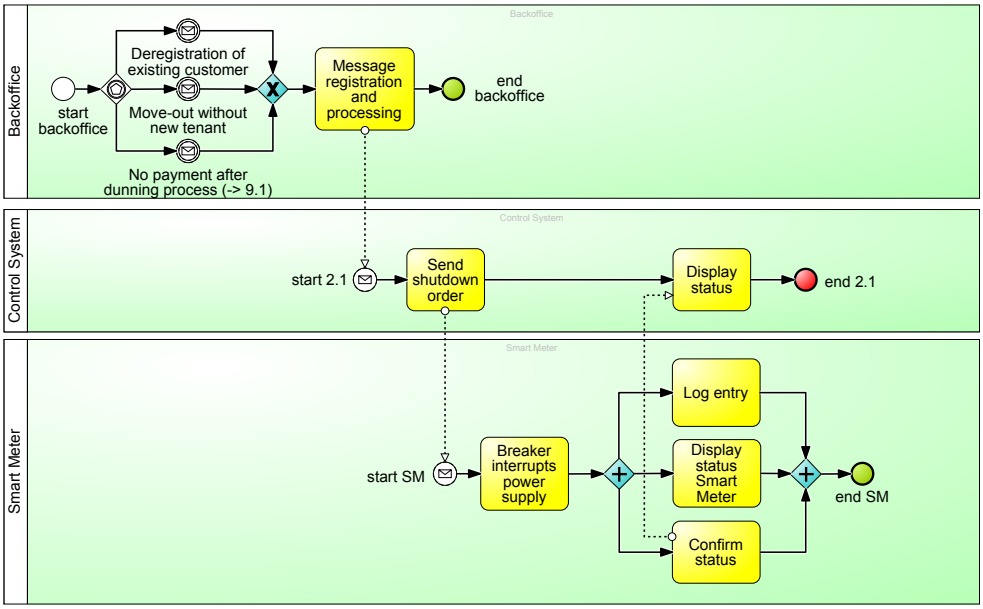

Figure 1: Use case 2.1 Shutdown after dunning or customer change.

shutdown process is investigated. These traces should be more realistic than traces that are created by transformations of valid traces that e.g. perform a skipping or swapping of arbitrary activities. On one hand the smart meter shutdown process (Figure 1) is modeled as realistically as possible on the basis of a textual description of smart metering use cases that the major energy providers in Austria have agreed upon (Oesterreichs-Energie, 2015). On the other hand possible attacks aiming at a shutdown are derived in a systematic way and the consequences of these attacks on the resulting traces are determined. Thus, the resulting traces should be highly realistic and could be further used for intrusion detection testing scenarios.

3 DERIVATION AND DETECTION OF SHUTDOWN ATTACKS

In this section attacks on a rather simple, but very important process, the shutdown process, are derived. Then their effects on the process are explored which determines how they could be detected. Both attacks and detection methods are modeled as an attack-defense trees. The consequences of the attacks on the traces give indications, how attacks could be simulated more realistically.

In order to simplify the analysis, the intrusion is assumed to happen at only a single location. In order to properly perform conformance checking we assume a dedicated intrusion detection system that collects all events and stores them in a log. Each event is stored together with the id of its process instance. Additionally, it is assumed that the creation and storage of the logs is unaffected by the intruder and that the information about the process is available for the analysis.

3.1 Smart Meter Shutdown Process

The process model of the remote smart meter shutdown use case was created based on a smart metering use case document (Oesterreichs-Energie, 2015). This document contains textual descriptions of smart metering use cases that the major energy providers in Austria have agreed upon. Use case 2.1 was identified as the use case that models the remote turn-off due to a customer deregistration or a moving out without a new tenant. Based on the textual description of the use case the shutdown process was modeled using the software tool Signavio.

The resulting process is shown in Figure 1. It starts with the reception of an incoming message that, e.g., states that the customer moved out. The message is registered in the backoffice which then sends a shutdown order to the control system which in turn sends a shutdown order to the smart meter. After interrupting the power supply, the smart meter sends its (new) status information back to the control system.

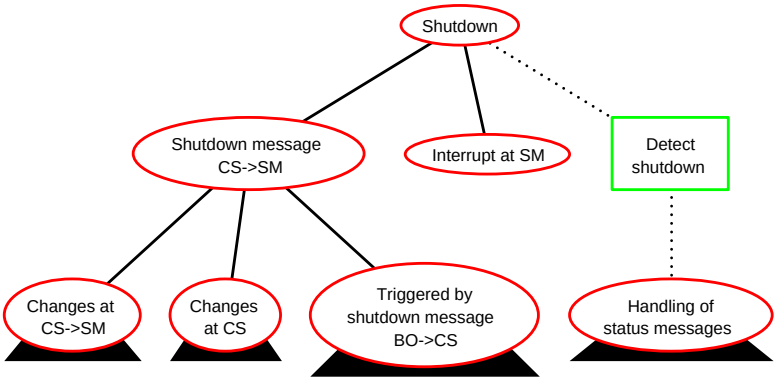

Figure 2: Top levels of the attack-defense tree.

3.2 Choice of Analysis Methodology

Both attack countermeasure and attack-defense trees (ADT) are general enough for modeling the trees of this paper. Attack defense trees were chosen since a software tool, called ADTools (Kordy et al., 2013), exists that allows easy creation and editing of the trees. Attacks are represented by red, elliptic nodes, detection measures are shown as green rectangles (Figure 2). All figures were created from the same attack-defense tree by hiding the parts above or below several nodes, which is indicated by black triangles above or below the corresponding nodes.

The considered attacks aim at forcing an irregular shutdown of smart meters. In the following, attacks achieving a shutdown are systematically constructed as an attack tree by starting with the activity "Breaker interrupts power supply" and going back along the process.

3.3 Attack-Defense Tree Derivation

In order to realistically describe effects of attacks, in this section attacks on the shutdown process are derived. Then their effects on the process are explored which determine (i) how they could be detected and (ii) how attacks could be simulated more realistically.

3.3.1 Top of the tree

The first level of the tree (Figure 2) corresponds to two different final attacks that can lead to a shutdown at the smart meter: (i) a shutdown message is sent from the control system (CS) to the smart meter (SM) and (ii) direct interruption at the smart meter (Figure 1).

The direct interruption at the smart meter (after an intrusion to the smart meter) can not be detected before the shutdown occurs by process mining which is indicated by the fact that no countermeasure node is below the node *Interrupt at SM*.

However, besides the two attacks, the first level also shows the detection method *Detect shutdown* which works independently from the attacks by detecting the shutdown itself through consideration of the status message that is sent *after* the shutdown took place. This detection method is described in more detail in Section 3.3.5 and depends on the way the attacker handles status messages.

Going one step back in the process model (Figure 1) a shutdown message from the control system (CS) to the smart meter (SM) can be achieved in three different ways: either by changes (i) at the communication line CS→SM, (ii) its origin CS or (iii) the part before (Triggered shutdown message). In the following, each of these 3 subgoals and its detection methods are described in more detail.

In order to execute some of the following detection methods, it is assumed that each message is expanded to include also the ID of the underlying process and the ID of the smart meter that should be turned off.

3.3.2 Shutdown Message CS→SM

A shutdown message from the control system to the smart meter can be achieved by replaying an old shutdown message or by creating a new shutdown message (Figure 3). A *replay* attack can be directly achieved by intercepting, possibly redirecting and sending a shutdown message. It can be detected by checking, if the received process ID is the ID of a currently active shutdown process. A redirection could additionally be detected when the receiving smart meter checks, if the shutdown message is including its ID. A replay to the same receiver can be additionally detected by comparing the received process ID with all previously obtained process IDs.

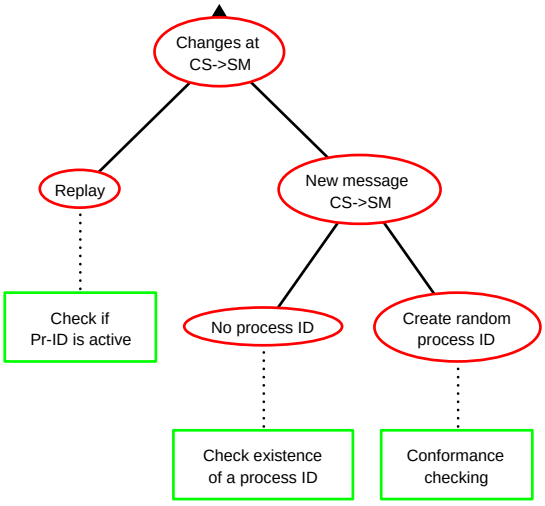

Figure 3: Attacks aiming at creating a shutdown message from the control system (CS) to the smart meter (SM) at the channel between them and the corresponding detection methods.

The second possibility is to create an entirely *new shutdown message* and send it to the smart meter. An intruder unaware of the underlying processes will not include a process ID, which can be detected immediately by checking the existence of a process ID.

A process-aware intruder could try to overcome the problem by creating a random process ID. If the created ID is completely new, conformance checking would detect the situation by recognizing a process which can not be a valid instance of the process model because it starts with the interrupt. If the created ID corresponds to a finished process or an active process in the wrong state (or more generally with an incompatible marking) the shutdown activity is not allowed (enabled) and the situation can be detected by conformance checking.

A general property of the methodology can already be seen at this stage of the analysis: the intrusion itself is not detected. Instead, the consequences of the intrusion when aiming at the shutdown such as missing or wrong IDs or wrong "states" of the system are detected. As such this approach works at a higher (process) level and is complementary to methods that detect the intrusion into the system.

It should be noted that the tree can be expanded at the attack leaves by intrusion goals. For example, a direct shutdown message from CS to the SM can be sent after either an intrusion to the CS or an intrusion to the communication channel between the CS and the SM. This was in fact done in order to test the analysis capability of ADT. By choosing the domain "satisfiability of the scenario" in ADT, it can be evaluated, if

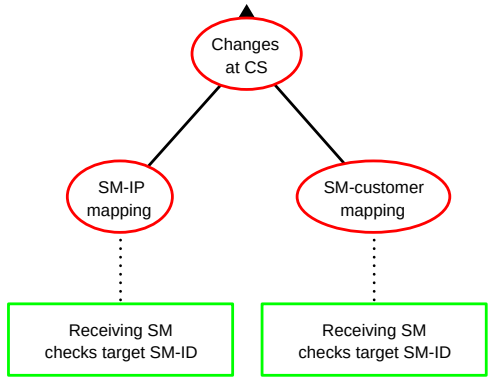

Figure 4: Attacks aiming at creating a shutdown message from the control system (CS) to the smart meter (SM) by changes at the CS and the corresponding detection methods.

an undetected shutdown occurs or not given that different intrusions, attacks and detection methods take place. Since the intrusions needed are rather obvious, this expansion is not done for sake of readability of the tree.

3.3.3 Changes at the Control Center

Many malicious changes at the control center are possible. It is plausible that the information about the linkage between smart meters, smart meter IDs and customers is stored at the control system and not in the backoffice. A modified mapping between smart meters and their IDs results in shutdown messages that are sent at the wrong smart meters (Figure 4). This can trivially be detected if the receiving smart meter compares the ID appended at the message with its own.

A modified mapping between smart meters and customers also results in shutdown messages that are sent at the wrong smart meters. Consequently it can be detected by the same method.

3.3.4 Shutdown Message BO→CS

After an intrusion in the corresponding communication channel, the shutdown message from the backoffice to the control system can be achieved either through a replay or through the creation of a new message (Figure 5).

The replay has already been discussed in Section 3.3.2 and can be detected with the same method. The new message can be achieved (i) by creating and sending a new message, (ii) by two changes at the backoffice of (iii) by an external message to the BO. The first possibility and its detection method has also already been discussed in Section 3.3.2.

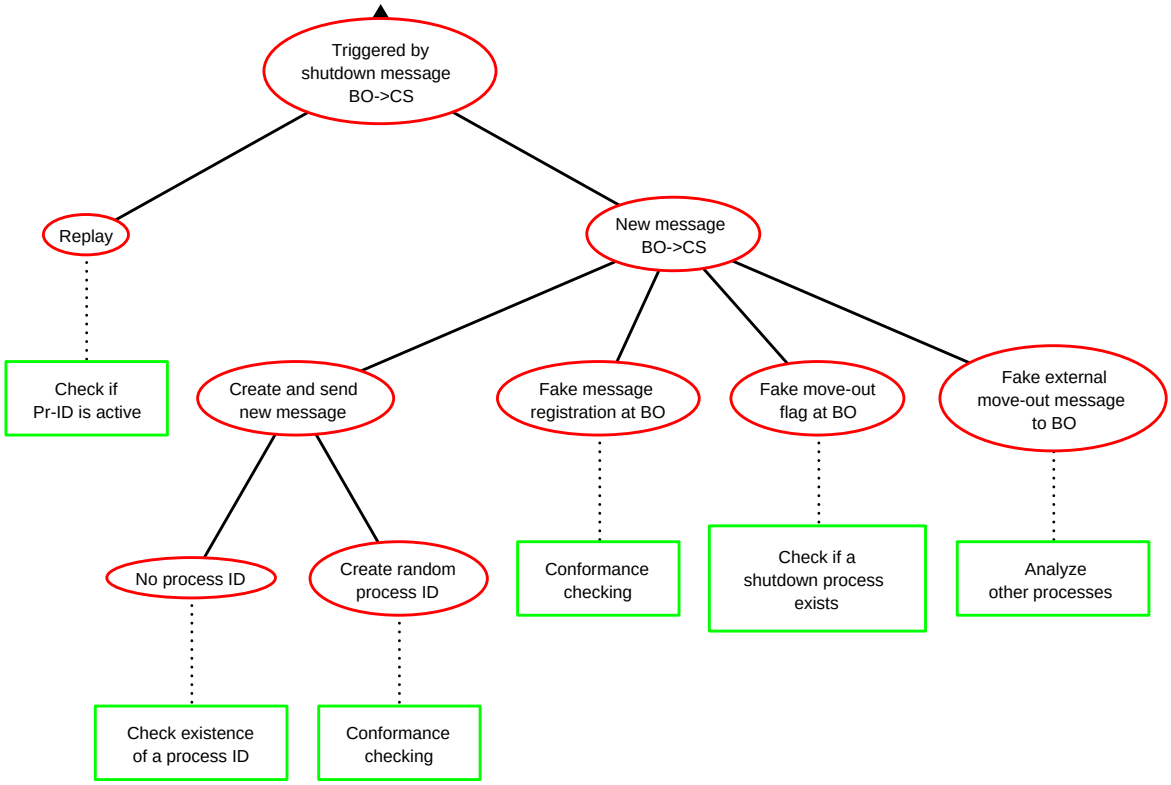

Figure 5: Attacks aiming at creating a shutdown message from the backoffice (BO) to the control system (CS) and the corresponding detection methods.

The next two attacks assume an intrusion in the backoffice. If the registration of, e.g., a move-out message is faked, a process ID is created and a shutdown message is sent to the CS. Since the incoming message event is not contained in the log, the corresponding trace is invalid. Therefore conformance checking would recognize this attack.

An even simpler attack could try to start the shutdown process by declaring that the customer has moved out. In the simplest case this attack could consist of changing a single boolean entry in a database. Note, however, that the modeled process needs to start with one of three incoming messages from the customer (denoted as double-circled events in Figure 1). Therefore, due to the absence of an incoming start message the shutdown process does not start. Consequently, the change of a boolean entry does not suffice for triggering a shutdown since no shutdown process exists. It should, however, be noted that in less strict processes such a change may suffice to trigger a shutdown.

Finally, the next attack considers the first step of the process and of sending a move-out message from outside to the backoffice. Although this attack is out of scope of this paper since no intrusion is necessary, it is interesting to study the implications of this attack. Since the reception of the message at the backoffice starts the shutdown process, it clearly can not be detected with process mining of the shutdown process. It turns out that the activity message registration and processing is itself more a process than a simple activity. Therefore, it should be modeled as a business process which could enable the detection of this non-intrusive attack.

3.3.5 Handling of Status Messages

So far, the part of the process before the shutdown has been considered. Except the direct shutdown after intrusion into the smart meter, in principle any of the detection methods described above could detect the attack before shutdown takes place. Now the part of the process before the shutdown has been considered. While the study of this process can not prevent a shutdown, it could still be used to detect the intrusion in order to e.g. prevent a large-scale shutdown.

Corresponding to the modeled process, after interrupting the power supply the smart meter sends its new status information back to the control system. While the attacker has reached its shutdown goal, he

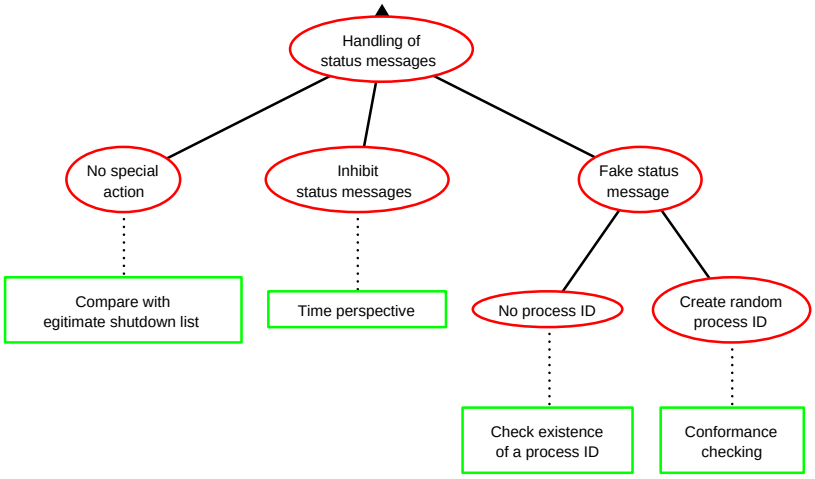

Figure 6: Different ways the attacker can deal with status messages and corresponding detection methods.

could try to remain undetected. The attacker has three possibilities (Figure 6).

If the attacker does nothing, the status message will be sent to the control center. Various ways to detect the illegitimate shutdown are possible. In the simplest case, a comparison with a simple list of smart meters that should have received a remote turn-off command will reveal the attack. For example, this happens for the so far undetected shutdown performed directly at the smart meter.

The simplest countermeasure of the attacker would consist of preventing the smart meter from sending status messages. Depending on the details of the system, this could be detected by the absence of regular status messages, i.e., by the consideration of the time perspective. The resulting never-ending process could for example be detected by considering the duration between the sending of the shutdown message from the CS to the smart meter and the reception of the status message.

Finally, the attacker could try to create and send a new status message. This possibility is analogous to the situation in Section 3.3.2 and can be detected as described there.

4 DISCUSSION

In this section, several general properties of the discussed detection approach are discussed. The discussion should clarify the benefits but also the limits of intrusion detection by process mining when done in the proposed way.

4.1 Consequences for Simulations of Attacks

This analysis above has consequences for testing the detection of intrusions by simulations: intrusions can lead to traces where the first activities are skipped, i.e., tasks from the beginning are missing (e.g., when the attacker creates and sends a new shutdown message). Moreover, the inhibition of sending the log messages will result in process instances that do not end. Therefore, while simulations that skip *single* tasks or exchange two tasks may be adequate for anomaly detection (Bezerra et al., 2009; Bezerra and Wainer, 2013), they are unrealistic and thus not suited for simulations of intrusions.

4.2 Consequences for Preprocessing

While never-ending traces also need to be considered in the testing simulations, they also need to be taken into account by the detection method. Some anomaly detection methods apply a scoping step in the preprocessing phase, where only finished processes remain (Bezerra et al., 2009). While scoping can be beneficial for anomaly detection, in the *intrusion* detection scenario, scoping will throw away the never-ending process instances which consequently lead to reduced intrusion detection rates.

4.3 Need for Other Perspectives

The same attack also reveals the need to take other perspectives like the time perspective into account considering, e.g., the duration between events. It is likely that the organizational view plays an important

role especially in business processes like, e.g., the registration and processing of incoming messages in the backoffice. The organizational perspective could also assist in the detection of the wrong connection between customers and smart meters. The data perspective could help in clarifying the smart meter status by, e.g., analyzing the energy consumption. As already stated above, intrusion detection based on the control-flow is more likely able to detect intrusions when the intrusion is *exploited* by a malicious attacker. In contrast, by definition, an eavesdropping attack does not change the way the system works. Therefore, the control-flow is unchanged and other perspectives are necessary for the detection of eavesdropping attacks.

4.4 Prohibition of the Shutdown

In order to prevent the shutdown, conformance checking would have to be done before the breaker interrupts. This requirement would also result in the need to analyze incomplete traces. In order to prohibit a large-scale turnoff due to an intrusion, conformance checking should be applied early enough, i.e., before the shutdown is performed, asking for a check by the intrusion detection system. Performing all these checks in time is suspected to require big efforts. However, even in this case the intrusion into the smart meter and its exploitation can not be detected before the shutdown by the consideration of the control flow.

4.5 Strongly Controlled Processes

The detection methods above either look at the process ID, the smart meter ID or perform conformance checking. While the check of the smart meter ID is clearly beneficial, the check of the process ID has a significant drawback: (shutdown) messages without a valid process-ID automatically need to be considered as dangerous. Therefore, the process must be strongly modeled, controlled and monitored not allowing unattributed messaged, i.e., messages that are not assigned to a certain process. Otherwise a lot of false positive alarms would occur. This property certainly limits the practical application. In order to work in practice, a way to distinguish messages that are not assigned to a process due to intrusions from unassigned messages due to loose process control will need to be found.

4.6 Limitation of the Analysis

The analysis above focuses on exploitations of intrusions that lead to a shutdown. Other attacks are also possible. As an example, an attack could try to confuse the system by inhibiting the smart meter from sending the confirmation of the status, send wrong status or consumption messages. While these attacks do not lead to a turnoff it could still result in considerable damage. For example wrong consumption values sent in another use case could result to a wrong system status which in turn could trigger wrong activities by the network operator.

5 Conclusion and Outlook

In this paper the potential of process mining for the detection of attacks on the smart metering shutdown use case is explored. This is done by systematically deriving attacks on the modeled shutdown process. For each attack, its detection using process mining is analytically explored and modeled by an attack-defense tree. It could be shown that process mining has the potential to detect exploitations of intrusions aiming at an illegitimate shutdown of smart meters. Based on this analysis several benefits and limitations of the method are discussed: from the methodical view, the control-flow oriented analysis should be accompanied by considering other views of the process. From a practical view, the fact that attacks can lead to unattributed messages is supposed to be problematic for loosely controlled processes.

Subsequent work will explore ways to combine the control flow analysis with other views. In addition to this analytic study, a comparative validation study with real data will be performed. Besides finding solutions for dealing with unattributed messages, the method will be compared with other approaches in terms of detection and false positive rates. This will be achieved by using an envisaged toolchain which is suited for an automated large-scale evaluation of the approaches.

REFERENCES

Accorsi, R. and Stocker, T. (2012). On the exploitation of process mining for security audits: the conformance checking case. In *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, pages 1709–1716. ACM.

Berthier, R., Sanders, W. H., and Khurana, H. (2010). Intrusion Detection for Advanced Metering Infrastructures: Requirements and Architectural Directions. In *2010 First IEEE International Conference on Smart Grid Communications*, pages 350–355. IEEE.

Bezerra, F. and Wainer, J. (2013). Algorithms for anomaly detection of traces in logs of process aware information systems. *Information Systems*, 38(1):33–44.

Bezerra, F., Wainer, J., and Van Der Aalst, W. M. P. (2009). Anomaly Detection using Process Mining. In *10th International Workshop, Enterprise, Business-Process and Information Systems Modeling*, volume 29, pages 149–161.

Jalali, H. and Baraani, A. (2012). Process aware host-based intrusion detection model. *International Journal of Communication Networks and Information Security*, 4(2):117–124.

Kordy, B., Kordy, P., Mauw, S., and Schweitzer, P. (2013). ADTool: Security Analysis with Attack-Defense Trees. In *International Conference on Quantitative Evaluation of Systems*.

Kordy, B., Mauw, S., Radomirović, S., and Schweitzer, P. (2012). Attack-Defense Trees. *Journal of Logic and Computation*, page exs029.

Kordy, B., Piètre-cambacédès, L., and Schweitzer, P. (2014). DAG-Based Attack and Defense Modeling : Don' t Miss the Forest for the Attack Trees. *Computer science review*, 13:1–38.

Oesterreichs-Energie (2015). Österreich Use-Cases für das Smart Metering Advanced Meter Communication System (AMCS).

Roy, A., Kim, D. S., and Trivedi, K. S. (2012). ACT: Towards unifying the constructs of attack and defense trees. *Security and Communication Networks*, 5(8):929–943.

Salter, C., Saydjari, O. S., Schneier, B., and Wallner, J. (1998). Toward A Secure System Engineering Methodology. In *Proceedings of the 1998 Workshop on New Security Paradigms (NSPW '98)*, pages 2–10.

Stocker, T. and Accorsi, R. (2013). SecSy: Security-aware Synthesis of Process Event Logs. In *Proceedings of the 5th International Workshop on Enterprise Modelling and Information Systems Architectures*, St. Gallen, Switzerland.

Van der Aalst, W. M. (2011). *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer.

Van der Aalst, W. M. and de Medeiros, A. K. A. (2005). Process mining and security: Detecting anomalous process executions and checking process conformance. *Electronic Notes in Theoretical Computer Science*, 121:3–21.