What's under the hood?

Improving SCADA security with process awareness

Justyna J. Chromik
University of Twente
Enschede, The Netherlands
j.j.chromik@utwente.nl

Anne Remke
University of Münster
Münster, Germany
anne.remke@uni-muenster.de

Boudewijn R. Haverkort
University of Twente
Enschede, The Netherlands
b.r.h.m.haverkort@utwente.nl

Abstract—SCADA networks are an essential part of monitoring and controlling physical infrastructures, such as the power grid. Recent news item show that tampering with the data exchanged in a SCADA network occurs and has severe consequences. A possible way of improving the security of SCADA networks is to use intrusion detection systems. By monitoring and analysing the traffic, it is possible to detect whether information has a legitimate source or was tampered with. However, in many cases the knowledge of just the traffic is not enough. Detecting intrusions could be improved by including awareness about the physical processes that are controlled. This paper shows a simple analysis of a small scenario of a power distribution system, to illustrate the benefits of including the knowledge about the process in detecting breaches in SCADA.

I. INTRODUCTION

Critical infrastructures, such as power generation and distribution systems, are crucial in every developed country. To ensure dependable operation of these systems, a reliable control network is needed. Geographically distributed systems are most often monitored and controlled by Supervisory Control And Data Acquisition (SCADA) networks. These networks were designed to ensure the availability of the system and used to: (i) be physically isolated from the corporate network and the Internet; (ii) use proprietary technologies; and (iii) not draw much attention. These three factors made them relatively secure.

Today, however, the information gathered by SCADA systems is often processed in corporate networks, e.g. to predict whether the current infrastructure is sufficient for future power demand. As a result, the physical isolation (*air gap*) is no longer there. Moreover, the proprietary protocols and technologies have been replaced with popular, standardised, or commercial off-the-shelf (*COTS*) solutions. Finally, with the examples of Stuxnet [1] or the Ukrainian grid hack [2], we can no longer believe that industrial systems do not attract attention.

Due to the quite different characteristics of SCADA networks as opposed to regular IT networks, e.g., corporate networks, the approaches used in IT networks are not usually applicable to SCADA networks. Moreover, it has been found many times that it is infeasible to create a completely safe network. For this reason, a possible way of improving the security of SCADA networks is by implementing an intrusion detection system

(IDS), and proposing countermeasures to the detected breaches. While many IDSs have been proposed in the literature [4], [5], only few are actually taking into account the physical process that is controlled [6], [7]. Bigham et al. [6] only focus on dependencies between the values of power in different parts of the power grid, whereas Cardenas et al. [7] addresses the physical properties of a water tank. Both of the approaches show the benefits of the process awareness, therefore we decide to follow and investigate further the approach of Bigham et al. in intrusion detection in power systems.

As said we believe that taking the physical process into account in more detail is crucial to propose an effective IDS for SCADA networks. Therefore, this paper illustrates the possibilities that arise when explicitly considering the interfaces between the SCADA network and the physical process, i.e., the state and readings of sensors and actuators. Even though several standardised IEEE power cases [8] exist, we present a different basic scenario of a distribution network that includes the connection to the SCADA network, as well as the SCADA network, explicitly. We outline an algorithm that is able to detect undesired commands and non safe or non consistent system states by checking several standard requirements for power grids and physical constraints, e.g. Kirchhoff's law. Even though the case presented and the algorithm outlined do not take into account the full complexity of the system, this paper shows already the capabilities of taking into account the state of the physical process.

This paper is organised as follows. Section II defines the system and system state, based on a small example of a power distribution system. Section III explains the approach for monitoring the system state. In Section IV we provide three examples for monitoring such a system. We conclude the paper in Section V.

II. THE CONTROL OF POWER DISTRIBUTION

In the following, we show by means of an example in what ways the power distribution system is controlled by the SCADA network. Section II-A introduces said example and Section II-B formalises the description of a system at hand. Section II-C defines the state of the system, and Section II-E lists requirements and physical restrictions which ensure the well-operation of the system.

Anne Remke is also a part time employee at the University of Twente, Enschede, The Netherlands.

A. Informal system description

Power distribution systems are large and complex systems of connected power lines, buses, switches, transformers, etc., which connect the sources of power to its consumers. To ensure a reliable operation, the power distribution is controlled by a SCADA network (control network). An example of SCADA network is shown in Figure 1. This control network manages the physical infrastructure:

- it measures values of current and voltage in power lines by means of *sensors*, e.g., voltage meters;
- it controls the physical infrastructure with *controllers*, either through a logical program, or by an operator;
- it applies changes by means of *actuators*, e.g., switches.

The data from sensors is collected by Remote Terminal Units (RTUs), Intelligent Electronic Devices (IEDs), or Programmable Logic Controllers (PLCs), and is sent to a Master Terminal Unit (MTU) located in the control network. RTUs, IEDs and PLCs are connected to the control room by means of some communication link, which is typically fibre or GPRS. Commands are sent from the control network back over the communication channel to the actuators. Some systems also have local control loops, which operate autonomously. The information collected in the field is stored in a SCADA server or historian, but it is also processed and displayed on the Human Machine Interface (HMI), which the operating engineer uses to get an overview of the system. One of the uses for SCADA is estimating the state of the system. These computationally heavy procedures estimate the state and possible future states of the system. This way the system is able to predict, for example, whether additional power needs to be provided to the grid. Inspired with the state estimators, we would like to investigate how similar analysis can improve the security of SCADA system. The control network is also connected to the corporate network of the company, as shown in Figure 1. This connection is one of the main risk factors for SCADA security. With a wrong configuration, anyone connected to the Internet can potentially connect to the control network.

B. Formal system description

Figure 2 presents a basic scenario that represents a small part of a low voltage power distribution system, which is used in the following to illustrate the interaction of the control network and the physical system. The scenario is inspired by power flow problems [8], and enhanced with sensors and switches. We denote the system description in the following way: the system Ω is a tuple $\Omega = (P, B, L, S, M)$, where $P = P^G \cup P^L$ is a set of power sources (P^G) and loads (P^L), B is a set of buses, L is a set of power lines, S is a set of switches, and M is a set of sensors (measuring the voltage and the current in the lines). Power sources and loads are kept as one vector to allow simpler state representation later on. These elements are connected with each other forming a certain topology. Note that, this paper does not formally describe the topology, yet, and we refer to a graphical description of the topology, as presented in the Figure 2.

Fig. 1: Basic topology of SCADA network.

Blue lines in Figure 2 belong to the SCADA system; this control network is able to: (i) read the sensor readings, (ii) control the switches, and (iii) know the state of the switch. We denote connections to switches and sensors separately in Figure 2. For the time being we assume that measurements are taken at the beginning, and at the end of each power line. Moreover, we assume that there is a switch at the beginning and end of each power line.

Black horizontal lines in Figure 2 represent power lines, labelled l_i for $i \in \{1, \dots, |L|\}$, and black vertical lines represent buses, labelled b_j for $j \in \{1, \dots, |B|\}$. We assume that the resistance of power lines is zero, therefore power loss on the power lines is negligible. The scenario presents a low voltage setting, and there are no backup buses present. Due to physical constraints, each power line has some maximum current capacity $I_{l_i}^{max}$ which, when exceeded, damages the power line. The maximum current capacity is written in red next to each power line in Figure 2, but can also be written as a vector $I_{\Omega}^{max} = \{I_{l_1}^{max}, I_{l_2}^{max}, \dots, I_{l_{|L|}}^{max}\}$ of size $|L|$. Each power line has a switch at each connection with a bus. We denote the switch as sw_{ij} , where $i \in \{1, \dots, |L|\}$ indicates the number of the power line the switch is placed on, and $j \in \{1, \dots, |B|\}$ indicates the number of the bus it is located next to. We denote the state of switch sw_{ij} as $ST_{sw_{ij}} \in \{0, 1\}$, representing an open (disconnected) and closed (connected) switch, respectively. The vector SW collects the states of all the switches and is of size $|S|$. Next to the switches each power line has sensors situated close to the bus. The sensors measure usually at least the current in the line $I_{l_{ij}}$, and the

Fig. 2: The basic power distribution scenario: black lines represent the physical system, blue lines represent the communication network.

voltage between the line and the ground $V_{l_{ij}}$. The subscript ij denotes the position at line i , and bus j for $i \in \{1, \dots, |L|\}$ and $j \in \{1, \dots, |B|\}$. The description of the readings from one sensor can be written as a pair of the current and voltage readings: $(I_{l_{ij}}, V_{l_{ij}})$.

The power flows from sources to loads. In standard distribution this is always the same direction; this enables us to note down for each bus subsets of “incoming” and “outgoing” lines. Let In_{B_k} denote the subset of lines incoming to the k th bus, and Out_{B_k} the subset of lines outgoing from the k th bus. For the scenario in Figure 2, we obtain the following subsets: $In_{B_1} = \{l_1, l_2\}$, $In_{B_2} = \{l_3, l_4, l_5\}$, $Out_{B_1} = \{l_3, l_4, l_5\}$, and $Out_{B_2} = \{l_6, l_7, l_8, l_9\}$. Note, that in smart grids there may be customers (houses) that produce electricity using, e.g., solar panels and are able to become a source of electricity. In that case they can be treated as a source, instead of a load. In Figure 2, the circles on the left with G inside represent sources of power, whereas the circles on the right with L inside represent loads. The amount of power produced at source G_i is denoted P_{G_i} and the amount of power consumed at load L_i is denoted P_{L_i} , respectively. We restrict ourselves to constant power loads and sources. Vector P is of size $|P|$ and collects the current values for all loads and sources. Note that loads are presented as negative values to distinguish them from (positive) source values.

C. System state

Let us now formalise the description of a state in the system depicted in Figure 2. The state refers to the values which can change in the system over time. The system state can be described by three vectors indicating: (i) the states of the switches, (ii) the sensor readings, and (iii) power consumption and production. Firstly, we denote the state of all switches in the system as a vector $SW = [ST_{sw_{11}}, \dots, ST_{sw_{|L||B|}}]$ which is of size $|S|$. Secondly, the readings from one sensor can be written as a pair of the measured current and voltage: $(I_{l_{ij}}, V_{l_{ij}})$. Vector SR collects those pairs for all sensors: $SR = [(I_{l_{11}}, V_{l_{11}}), \dots, (I_{l_{|L||B|}}, V_{l_{|L||B|}})]$, and is of size $|M|$. Finally,

the vector describing the loads and sources of power is denoted as $P = [P_{G_1}, \dots, P_{G_{|P^G|}}, -P_{L_1}, \dots, -P_{L_{|P^L|}}]$ for $|P^G|$ sources and $|P^L|$ loads. In this work we assume constant values of sources and loads; we want to extend this to variable values in future. Now, the system state T can be written as a tuple that consists of the above three vectors: $T = (SW, SR, P)$.

D. Events

Having the system state described, let us now consider which events can change the state of the system. Any new information received about the system can possibly change the system state we are examining: e.g. information from the sensors about changing the voltage reading from 230 V to 232 V is a change in the state. Different power values of the sources or loads also bring us to a new state. Moreover, a command to open or close any of the switches brings the system to another state. Since we assume constant power sources and loads, let us, for now, only consider two event types: (i) readings, and (ii) commands. Readings will refer to a new tuple $T' = (SW', SR', P')$, whereas a command will give us a new vector SW' .

E. Requirements and restrictions

To ensure a reliable and secure power distribution the system needs to fulfil a couple of safety requirements ($\mathcal{R}$) and physical restrictions ($\mathcal{P}$). The safety requirements need to be taken into account when designing and controlling the system, since any violation of these requirements may lead to damage in the system. The following **requirements** need to be ensured:

- $\mathcal{R}_1$. The voltage on all lines stays between the boundaries $230V \pm 10\%$, i.e. $V_{l_{ij}} \in [207; 253] \forall i, l_i \in L$.
- $\mathcal{R}_2$. The current in a power line does not exceed the maximum allowed current in that power line, i.e. $I_{l_i} \leq I_{l_i}^{max}$ for $\forall i, l_i \in L$.
- $\mathcal{R}_3$. All the loads are connected to some source of power. By this we only mean the connection between a house and a source of power, not considering the appliances within the house.

$\mathcal{R}_4$. The power produced by the sources equals the power consumed by loads, i.e. $\sum_{P_{G_i} \in P^G} P_{G_i} = \sum_{P_{L_i} \in P^L} P_{L_i}$. Requirements $\mathcal{R}_1$ and $\mathcal{R}_4$ directly follow from standards for power generation and distribution of electricity, e.g. [9]. Requirement $\mathcal{R}_2$ is defined by the producer of the cable, since it has physical limitations. Requirement $\mathcal{R}_3$ is in best interest of the distribution system operator, because they have to compensate their customers (power suppliers) financially when power outages occur. Note that in this paper the requirement $\mathcal{R}_3$ will not be dealt with, and will be addressed in future work, once the topology is formally defined.

Moreover, several **physical laws** exist that describe the relationship between different entities of the power grid:

- $\mathcal{P}_1$. The sum of current ingoing to a bus must be equal to the sum outgoing from a bus (Kirchoff's current law),

$$\forall B_i \in B. \left(\sum_{l_k \in In_{B_i}} I_{l_{ki}} = \sum_{l_j \in Out_{B_i}} I_{l_{ji}} \right).$$

 $\mathcal{P}_2$. If a switch on a line p is open, the values of current and voltage on this line have to be zero: $(ST_{sw_{pj}} = 0) \Rightarrow (\forall_{j \in [0, \dots, |B|]} ((I_{l_{ij}} = 0) \wedge (V_{l_{ij}} = 0)))$.
 $\mathcal{P}_3$. Assuming there is no power loss on the lines, the value of current and voltage on the beginning of the line must be equal to the value of the current and voltage at the end of the line, e.g. $I_{l_{pi}} = I_{l_{pj}}$ and $V_{l_{pi}} = V_{l_{pj}}$ for line p and buses i, j .
 $\mathcal{P}_4$. The electric power is equal to voltage times the current in the line $P = V \cdot I$.

The requirements $\mathcal{R}_1 - \mathcal{R}_4$ will be used to detect if a system moves to an undesirable state, especially after executing a command; the physical restrictions $\mathcal{P}_1 - \mathcal{P}_4$ will be used to determine if the current system is consistent when receiving a new reading, but they will also be used to calculate the predicted state of the system, once the new command is executed.

III. MODEL-BASED MONITORING

Monitoring the system state over time allows the system operator to (i) validate whether the system evolves consistently, and (ii) evaluate whether the execution of a command will lead to a safe system state in advance.

A. Approach

Monitoring is done by regularly checking the requirements $\mathcal{R}_1 - \mathcal{R}_4$ and the physical constraints $\mathcal{P}_1 - \mathcal{P}_4$. The execution of a command is considered to be unsafe if it leads to a system state that violates one of the requirements $\mathcal{R}_1 - \mathcal{R}_4$. On the other hand, if one of the physical constraints $\mathcal{P}_1 - \mathcal{P}_4$ is violated, this indicates that the information that is available on the system state must be incorrect.

The readings of sensors are not always correct due to different external factors such as: (i) a faulty sensor, (ii) a lost message in the communication channel, or (iii) injecting a false message into the system by a third party. The system operator controls the system based on the information that is provided by the SCADA system. Hence, it is of utmost importance to identify inconsistencies in the representation

of the system state. Therefore, as a next step we look for discrepancies between the system perceived by the operator, i.e., given by the readings from sensors, and the expected system state, described by physical laws governing the system. These discrepancies will determine whether the sensor readings are possibly wrong, or whether the issued command may harm the system, and alert the operator about this.

B. Outline of the algorithm

In Section II-C we defined the system state as a tuple $T = (SW, SR, P)$. Let us distinguish two system states: (i) T_O , which is the perceived system state, that the Operator sees by only analysing the sensor readings; and (ii) T_C , which is the Calculated system state obtained using the physical laws of the system. Note that in an ideal world the two system states, T_C and T_O , are the same. Upon an event, as defined in Section II-D, we would like to see whether the expected system state is consistent and safe, i.e., whether all the restrictions and requirements mentioned in II-E are met. The algorithm explaining this process is outlined in Figure 3.

The left part of Figure 3 shows the procedure taken when receiving new sensor readings. New readings mean that we have reached an entirely new system state $T'_O = (SW', SR', P')$, which could be unsafe and/or inconsistent. Therefore, two checks need to be performed: (i) the safety check, which is done by verifying the restrictions: $\mathcal{R}_1 - \mathcal{R}_4$, and (ii) the consistency check, which is done by verifying the physical constraints: $\mathcal{P}_1 - \mathcal{P}_4$. If the system is both consistent and safe, the reading T_O is simply replaced by T'_O . Otherwise an alert is generated with the reasons listed in Table I.

TABLE I: Alerts to a sensor reading

safe?	consistent?	alert
yes/no	no	system may be in danger or the information from sensor(s) is incorrect
no	yes	system is in danger: immediate reaction of system operator required

The right part of Figure 3 shows the action caused by a new command. When receiving a new command, i.e., a new vector SW' , this command is first "executed" in the model - based on knowledge of the current state T_C . If the predicted new state T'_C is safe, the command can be executed on the actual system, and the value T_C is updated to the current value T'_C . Otherwise, if the predicted state is unsafe, the command should be discarded and a proper alert has to be sent to the operator. This calculation is only meaningful if the system state is as expected, i.e. when $T_C = T_O$, otherwise we operate on data which is not necessarily correct.

The lower loop in Figure 3, compares the current state of the system T_O , as seen by the operator, to the previously calculated system state T_C . If these two states are not the same (with an error margin of ϵ), this has to be reported to the operator, since it indicates a potentially dangerous situation. The proposed algorithm cannot provide a meaningful prediction when the information it uses is incorrect. Therefore, the operator will be

Fig. 3: Flow chart illustrating the algorithm detecting undesired commands and not consistent states. Circles represent state, triangles represent events, hexagons are the control checks, and the alerts are shown as diamonds.

notified about the inconsistency until the situation is solved, i.e., until a faulty sensor is replaced or repaired.

IV. EXAMPLES

To illustrate the use of the proposed model, we will present three examples: (i) where we verify the requirements and physical restrictions based on an example; (ii) where we can show that based on a new reading we discover a faulty sensor; and (iii) where based on a new command we are able to determine that the new state is unsafe and therefore the command should be discarded.

A. Normal operation

Let us now consider a system like in Figure 2, where the maximum currents are given as $I_{\Omega}^{max} = [0.8, 0.5, 0.3, 0.4, 0.5, 0.3, 0.5, 0.3, 0.3]$. Moreover, let us assume that the initial state of the system depicted is $T_O = T_C = (SW, SR, P)$, where:

$$\begin{aligned} SW &= [ST_{sw11}, \dots, ST_{sw92}] \\ &= [1, 1, 1, 1, 1, 1, 1, 1, 1, 1], \\ P &= [P_{G1}, P_{G2}, -P_{L1}, -P_{L2}, -P_{L3}, -P_{L4}] \\ &= [100, 50, -20, -80, -30, -20]. \end{aligned}$$

The sensor readings, and therefore the SR vector are given in Table II, under “Reading 1”. Taking the initial vector SW as the command SW' , we can compute the initial T_C for constant power sources and loads defined in P . We can then evaluate whether the requirements and physical constraints are met:

- $\mathcal{R}_1$. Voltage, as measured by sensors in all power lines is in between the boundaries $V_{11}, \dots, V_{92} \in [207; 253]$ V (see also Table II under “Reading 1”).
- $\mathcal{R}_2$. The currents, as measured by sensors in all power lines do not exceed the maximum current, e.g. $I_{11} = 0.43 < 0.8 = I_{11}^{max}$ (see also Table II under “Reading 1”).
- $\mathcal{R}_3$. The loads are connected to sources of power.
- $\mathcal{R}_4$. The sum of powers is equal to zero: $100 + 50 - 20 - 80 - 30 - 20 = 0$.
- $\mathcal{P}_1$. This holds for all the buses, e.g. the sum of currents in bus 1: $0.43 + 0.23 = 0.22 + 0.22 + 0.22$.
- $\mathcal{P}_2$. All the switches are closed, so the constraint holds.

- $\mathcal{P}_3$. For the lines l_3, l_4 and l_5 we read that: $I_{l_{31}} = I_{l_{32}} = 0.22$, $I_{l_{41}} = I_{l_{42}} = 0.22$ and $I_{l_{51}} = I_{l_{52}} = 0.22$.

- $\mathcal{P}_4$. For all the sources and loads it holds that $P = V \cdot I$.

B. Faulty sensor

In this scenario we use the system state from Section IV-A, and assume that a new set of readings T'_O was received. The values of T'_O are presented in Table II in the column “Reading 2”. We assume that sensor 41 is broken, which leads to faulty reading $I_{l_{41}}$, what is depicted in **bold**. The new reading is analysed to determine if it meets the requirements $\mathcal{R}_1 - \mathcal{R}_4$ and the physical constraints $\mathcal{P}_1 - \mathcal{P}_4$. We see that the requirements $\mathcal{R}_1 - \mathcal{R}_4$ are still met, however, upon checking the physical constraints we discover that:

- $\mathcal{P}_1$. The sum of currents in bus 1 does not sum to zero: $0.43 + 0.23 = 0.66 \neq 0.79 = 0.22 + \mathbf{0.35} + 0.22$. This suggests that at least one of the sensors gave a wrong value: $I_{l_{11}}, I_{l_{21}}, I_{l_{31}}, \mathbf{I}_{l_{41}}$ or $I_{l_{51}}$.
- $\mathcal{P}_3$. For line l_4 we read that: $I_{l_{41}} = \mathbf{0.35} \neq 0.22 = I_{l_{42}}$. This suggests that at least one of the sensors gave a wrong value: $\mathbf{I}_{l_{41}}$ or $I_{l_{42}}$.

Even without knowing which sensor is broken, from these two observations we can conclude that the value of $\mathbf{I}_{l_{41}}$ is false. Note that the above algorithm is able to detect that the system state is different than expected. However, it cannot yet draw the conclusion as presented above. Once the model is extended with the topology, the algorithm can be adjusted accordingly, so that it will be able to detect which sensor is failing. The SCADA monitoring device can then issue a warning to the operator about a faulty reading; in case the operator receives many warnings, a conclusion can be drawn that the sensor is broken and needs to be replaced. The state T_O is updated with T'_O , and therefore, it will not be equal to the previously calculated T_C . The bottom loop in Figure 3 checking if the current $T_O = T_C$ will yield an alert notifying the operator about this fact.

C. Undesirable command

Let us now assume that the system is in state $T_O = T_C = (SW, SR, P)$. At one point, an event occurs with a command to open switch sw_{51} , i.e., resulting in the following vector

TABLE II: The maximum current on the line and the sensor readings

Line	I_i^{max}	Reading 1				Reading 2				Prediction			
		Bus 1		Bus 2		Bus 1		Bus 2		Bus 1		Bus 2	
		I_{ij}	V_{ij}	I_{ij}	V_{ij}	I_{ij}	V_{ij}	I_{ij}	V_{ij}	I_{ij}	V_{ij}	I_{ij}	V_{ij}
1	0.8	0.43	230	-	-	0.43	230	-	-	0.43	230	-	-
2	0.5	0.23	230	-	-	0.23	230	-	-	0.23	230	-	-
3	0.3	0.22	230	0.22	230	0.22	230	0.22	230	0.33	230	0.33	230
4	0.4	0.22	230	0.22	230	0.35	230	0.22	230	0.33	230	0.33	230
5	0.5	0.22	230	0.22	230	0.22	230	0.22	230	0	0	0	0
6	0.3	-	-	0.09	230	-	-	0.09	230	-	-	0.09	230
7	0.5	-	-	0.35	230	-	-	0.35	230	-	-	0.35	230
8	0.3	-	-	0.13	230	-	-	0.13	230	-	-	0.13	230
9	0.3	-	-	0.09	230	-	-	0.09	230	-	-	0.09	230

$SW^1 = [1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1]$. Following the outlined algorithm in Figure 3, we calculate the resulting T'_C , using the physical constraints $P_1 - P_4$. We see that line 5 does not conduct electric power, i.e. $I_{l_{51}} = 0$, $I_{l_{52}} = 0$, $V_{l_{51}} = 0$, and $V_{l_{52}} = 0$, as we know from physical constraint P_2 . Therefore, the power will have to be conducted by the two lines: l_3 and l_4 instead of the three lines: l_3 , l_4 and l_5 . From P_1 we know that the incoming currents are equal to outgoing currents, therefore $I_{l_{11}} + I_{l_{21}} = I_{l_{31}} + I_{l_{41}}$. The lines should be used evenly (we assumed that the resistance of power lines is zero), therefore $I_{l_{31}} = I_{l_{41}} = \frac{I_{l_{11}} + I_{l_{21}}}{2} = 0.33$. Moreover from P_3 we know that $I_{l_{32}} = I_{l_{31}} = 0.33$ and $I_{l_{42}} = I_{l_{41}} = 0.33$. We gather the results obtained this way in Table II under column “Prediction” (grey columns). We can see that the current in line l_4 exceeds the maximum allowed current in this line (dark grey cells in Table II). Therefore, the command has to be discarded, and an alert needs to be raised to the operator.

V. CONCLUSIONS AND FUTURE WORK

Operators of power distribution systems rely on SCADA networks to monitor and control these complex systems. False information about the system can lead to wrong decisions by the operator, potentially causing power outages or physical damage to the elements of the distribution grid. Therefore, we would like to know whether the sensor readings are consistent with our expectations or if they have possibly been tampered with. So far, securing SCADA networks was done without including the awareness of the underlying physical process. However, this paper illustrated the possibilities that arise when monitoring the state of the physical process and checking for consistency and safety, continuously.

This paper uses a small part of the power distribution network to illustrate the interaction between the physical process and the SCADA network. We formally describe the notion of a system state and outline the monitoring process, including the checks for safety and consistency, which are based on standard requirements and physical laws, respectively. Several examples show how this can be used to detect faulty sensor readings, as well as commands that lead to unsafe states. In its current form the main purpose of the algorithm is to alert the system operator. We expect that when formalising the topology, the algorithm can be extended to detect which sensors provide faulty information in many cases.

The scenario we presented is static in the sense that the power loads and sources do not change. We also apply a couple of simplifying assumptions, e.g., regarding the loss on power lines and the direction of the power flow. Further work will investigate how the presented approach can be extended to account for more complicated scenarios.

Clearly, the number of checks will grow with the system at hand. However, since the complexity of both, the consistency as well as the safety checks is constant in the number of components, we believe that monitoring is still feasible for a real system topology in real time. Many system operators already have an Energy Management System (EMS) in place, which is used for planning the power production and for monitoring the system as a whole. The capabilities of these EMSs vary a lot and to the best of our knowledge the information obtained is only used by process operators for control purposes. However, we have shown that the information present in such EMSs is of great value also on the network level and can be used to build intrusion detection systems that are tailored to the physical process at hand.

REFERENCES

- [1] N. Falliere, L. O. Murchu, and E. Chien, “W32. stuxnet dossier,” *White paper, Symantec Corp., Security Response*, vol. 5, 2011.
- [2] M. Assante, “Confirmation of a Coordinated Attack on the Ukrainian Power Grid.” <https://ics.sans.org/blog/2016/01/09/confirmation-of-a-coordinated-attack-on-the-ukrainian-power-grid/>, accessed January 10, 2016.
- [3] “The Repository of Industrial Security Incidents.” <http://www.risidata.com/>, accessed December 14, 2015.
- [4] B. Zhu and S. Sastry, “SCADA-specific intrusion detection/prevention systems: a survey and taxonomy,” in *Proc. of the 1st Workshop on Secure Control Systems (SCS)*, 2010.
- [5] R. Mitchell and I.-R. Chen, “A survey of intrusion detection techniques for cyber-physical systems,” *ACM Computing Surveys (CSUR)*, vol. 46, no. 4, p. 55, 2014.
- [6] J. Bigham, D. Gamez, and N. Lu, “Safeguarding scada systems with anomaly detection,” in *Computer Network Security*, pp. 171–182, Springer, 2003.
- [7] A. A. Cardenas, S. Amin, Z.-S. Lin, Y.-L. Huang, C.-Y. Huang, and S. Sastry, “Attacks against process control systems: risk assessment, detection, and response,” in *Proceedings of the 6th ACM symposium on information, computer and communications security*, pp. 355–366, ACM, 2011.
- [8] “Power Flow Cases.” <http://publish.illinois.edu/smartergrid/power-cases/>, accessed January 30, 2016.
- [9] “CENELEC Harmonisation Document: Nominal voltage for low voltage public electricity supply systems, HD 472 S1,” 988.