A Testbed for locally Monitoring SCADA Networks in Smart Grids

Justyna J. Chromik[1], Anne Remke[1,2] and Boudewijn R. Haverkort[1]

***Abstract*—This paper proposes a testbed for validating how our process-aware monitoring approach is able to increase the security of decentralized SCADA networks in power grids. The testbed builds on the simulation framework *Mosaik*, and co-simulates in an integrated way, the power distribution network itself, and the control network (Modbus/TCP). By extending the existing *Mosaik* framework with with a SCADA connection and possible topology changes, the testbed allows to investigate how our decentral monitoring approach increases security in distribution networks.**

***Index Terms*—SCADA, Bro, monitoring, process-aware, local monitoring, co-simulation, security**

I. INTRODUCTION

The ongoing integration of more renewable resources and new technology, like energy storage systems, into smart grids requires full integration of ICT into power transmission and distribution systems. To guarantee a stable power grid, many approaches propose Decentralized Energy Management (DEM), which relies on Supervisory Control and Data Acquisition (SCADA) networks to communicate sensor readings and commands between the individual components and their control server. While DEM remains a challenge, recent events, such as disconnecting the Ukrainian substations [1] through cyber attacks, have shown that also these control networks need to be improved w.r.t. their security and reliability.

One way to improve network security is to monitor ongoing traffic and to compare it to the current state of the system. Clearly, when doing this for larger networks one will suffer from scalability issues, hence, we aim at evaluating the value of decentralized monitoring approaches. We use the Bro intrusion detection system (IDS)[1] to monitor SCADA traffic together with state information of the underlying physical process to determine if the commands sent through the network are legitimate, as proposed in [2], [3]. By performing this monitoring in a local manner, we aim to detect malicious commands at remote substations, without involving a central control room. This will not only help to keep the decentralized energy management secure, but also avoid a centralized single point-of-failure, thus improving scalability.

This paper explains how we extend the co-simulation framework, *Mosaik*[2], that allows the simulation of the physical power distribution, with a discrete-event simulation of the Remote Terminal Units (RTUs) used for control purposes.

II. RELATED WORK

Combining information about the physical process being controlled with the state of the control network itself, has been investigated already, under different names: [4], [5] discuss this as semantic-based security analysis, [6], [7] as physics-based attack detection, while [8] refers to this as behavior-based.

Similarly to [5] our recent approach in [3] proposes to use measurements taken locally to create a semantic-aware distributed monitoring system with runtime verification. Since we aim to monitor and perform the detection analysis close to measurements, there is no need to simulate the entire control network, however we do require a simulation of an RTU connected to the SCADA network. In contrast, current co-simulation environments like [9], [10] focus on simulating the entire network, to analyze, e.g. denial of service attacks on the control network. These fully simulated approaches are highly flexible, while more advanced testbeds [7], [11], [12], may require a connection to emulate real hardware. Non-virtualized testbeds at Distribution System Operators are less flexible and often difficult to access. All simulation-based approaches require a power simulator, like Power World [10], [12], OpenDss [9] or Mosaik [13]. The latter easily integrates existing simulators in the smart grid co-simulation framework. This is the main reason, why we chose Mosaik and integrated (part of) the Modbus/TCP based control network and our monitoring tool, as shown in the following.

III. EXPERIMENTAL SET UP AND MONITORING

The testbed set-up as shown in Fig. 1 consists of: (i) *Mosaik*, the power grid simulator which determines the state of the power grid system; (ii) the control network, i.e., a Modbus RTU and a SCADA server; and (iii) Bro, a network monitoring and intrusion detection tool.

A. Mosaik

The extensible discrete-event co-simulation framework allows to connect various simulators. Our test set-up connects a simulator of the RTU, the power flows, the topology, and the

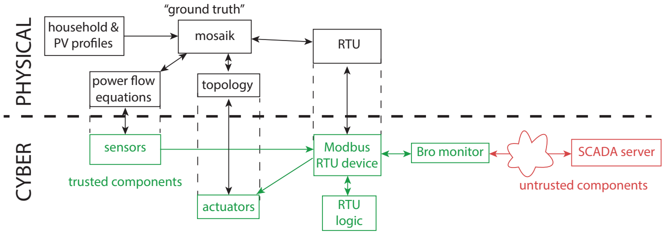

Fig. 1. Scheme of the testbed: simulated "ground truth" above the dashed line, and trusted (green) and untrusted components (red) below.

[1]Design and Analysis of Communication Systems, University of Twente, The Netherlands {j.j.chromik, a.k.i.remke, b.r.h.m.haverkort} at utwente.nl

[2]Safety-critical systems group, University of Münster, Germany anne.remke at uni-muenster.de

[1]https://www.bro.org/index.html

[2]https://mosaik.offis.de

household/workload and PV generation profiles. We adapted the power flow simulator available in Mosaik, which uses PYPOWER[3], to incorporate topology changes. As a result, the simulator is now able to output sensor readings for an *updated* topology (where e.g. a line in power distribution is opened or closed), i.e. the power flow solver computes the current, voltage and the phase angles of the power lines and buses in the system.

B. The Modbus/TCP SCADA system

In our example scenario, the Modbus/TCP SCADA system consists of one RTU and one SCADA server located in the control room, cf. Fig. 1. It uses the PyModbus library[4] implementing the Modbus/TCP protocol [14]. The RTU reads the measurements from the sensors directly connected within the substation and controls a set of actuators (switches) connecting power lines attached to the same bus. Additionally, it performs internal safety and security checks, based on which the RTU's logic decides whether to perform an action (e.g., open or close a switch). In case an action is required locally or from the central SCADA server, the RTU forwards the appropriate command to the respective actuator(s). Each instance of an RTU is created by the RTU simulator and controlled by Mosaik. It communicates over an untrusted network with the central SCADA server (which we assume to be an untrusted component as well because of possibility of insider attacks).

C. Bro Network Intrusion Monitor

Bro is a real-time network traffic monitor used, among others, for intrusion detection in SCADA systems [4], [15]. It includes a Modbus/TCP parser, that generates events upon parsing Modbus/TCP packets. For example, the *modbus_write_single_coil_request* event is generated when parsing a Modbus/TCP packet containing a "write single coil request". By creating a new event handler, we can instantiate new policies that use the semantic information extracted from the parsed packet to determine a proper action and alert. In this way we implement the monitoring approach proposed in [2], [3]. When detecting a command that can bring this system to an undesired state, Bro will detect such a command, and subsequently alerts, postpones or rejects its execution.

IV. CONCLUSIONS AND FUTURE WORK

The testbed presented here can be used to implement and evaluate local-based monitoring of SCADA systems as, e.g., proposed in our earlier work. We designed a monitoring environment, which co-simulates the power distribution simulation by Mosaik with a discrete-event network based on Modbus/TCP simulation, and the Bro monitoring IDS.

One of the challenges that we have encountered in using the developed testbed is performing the security checks defined in the RTU logic in real-time. Due to the internal discrete-step co-simulation approach in Mosaik, a non-negligible delay may occur before a change in one simulator is passed to another mutually dependent simulator. We will improve the synchronization of simulators in the future, by identifying dependencies and invoking dependent simulators after major changes.

Also, we will compare the detection performance of our local monitoring with a centralized approach [4]. Furthermore, we will investigate the minimal amount of local information necessary for the system to perform accurate monitoring. We plan to test our approach on the IEEE benchmark suite [16].

ACKNOWLEDGMENT

This research is funded through the NWO project (”MOre secure scada through SElf-awarenesS”) grant nr. 628.001.012.

REFERENCES

[1] ICS-CERT, "Alert (IR-ALERT-H-16-056-01) Cyber-Attack Against Ukrainian Critical Infrastructure." https://ics-cert.us-cert.gov/alerts/IR-ALERT-H-16-056-01, released February 25, 2016.

[2] J. J. Chromik, A. Remke, and B. R. Haverkort, "What's under the hood? improving scada security with process awareness," in *Joint Workshop on Cyber-Physical Security and Resilience in Smart Grids*, pp. 1–6, IEEE, 2016.

[3] J. J. Chromik, A. Remke, and B. Haverkort, "Improving scada security of a local process with a power grid model," in *4th International Symposium for ICS&SCADA Cyber Security Research 2016*, BCS Learning & Development Ltd., 2016.

[4] H. Lin, A. Slagell, Z. Kalbarczyk, P. Sauer, and R. Iyer, "Runtime semantic security analysis to detect and mitigate control-related attacks in power grids," *IEEE Transactions on Smart Grid.*

[5] A. Wain, S. Reiff-Marganiec, H. Janicke, and K. Jones, "Towards a distributed runtime monitor for ICS/SCADA systems," 2016.

[6] D. I. Urbina, J. A. Giraldo, A. A. Cardenas, N. O. Tippenhauer, J. Valente, M. Faisal, J. Ruths, R. Candell, and H. Sandberg, "Limiting the impact of stealthy attacks on industrial control systems," in *Proc. of ACM SIGSAC Conf. on Computer and Communications Security*, pp. 1092–1105, ACM, 2016.

[7] G. Koutsandria, R. Gentz, M. Jamei, A. Scaglione, S. Peisert, and C. McParland, "A real-time testbed environment for cyber-physical security on the power grid," in *Proc. of the 1. ACM Workshop on Cyber-Physical Systems-Security and/or PrivaCy*, pp. 67–78, ACM, 2015.

[8] H. Bao, R. Lu, B. Li, and R. Deng, "Blithe: Behavior rule-based insider threat detection for smart grid," *IEEE Internet of Things Journal*, vol. 3, no. 2, pp. 190–205, 2016.

[9] A. Awad, P. Bazan, and R. German, "Sgsim: Co-simulation framework for ict-enabled power distribution grids," in *Int. GI/ITG Conf. on Measurement, Modelling, and Evaluation of Dependable Computer and Communication Systems*, pp. 5–8, Springer, 2016.

[10] C. Davis, J. Tate, H. Okhravi, C. Grier, T. Overbye, and D. Nicol, "Scada cyber security testbed development," in *Power Symposium, 2006. NAPS 2006. 38th North American*, pp. 483–488, IEEE, 2006.

[11] B. Kang, P. Maynard, K. McLaughlin, S. Sezer, F. Andrén, C. Seitl, F. Kupzog, and T. Strasser, "Investigating cyber-physical attacks against iec 61850 photovoltaic inverter installations," in *20th Conf. on Emerging Technologies & Factory Automation*, pp. 1–8, IEEE, 2015.

[12] P. Gunathilaka, D. Mashima, and B. Chen, "A real-time testbed environment for cyber-physical security on the power grid," in *Proceedings of the 2nd ACM Workshop on Cyber-Physical Systems Security and Privacy*, pp. 113–124, ACM, 2016.

[13] F. Schloegl, S. Rohjans, S. Lehnhoff, J. Velasquez, C. Steinbrink, and P. Palensky, "Towards a classification scheme for co-simulation approaches in energy systems," in *Int. Symp on Smart Electric Distribution Systems and Technologies*, pp. 516–521, IEEE, 2015.

[14] "The Modbus Organization, Modbus application protocol specification, ver. 1.1b3," 2012.

[15] R. Udd, M. Asplund, S. Nadjm-Tehrani, M. Kazemtabrizi, and M. Ekstedt, "Exploiting Bro for intrusion detection in a SCADA system," in *Proc. of the 2nd ACM Int. Workshop on Cyber-Physical System Security*, pp. 44–51, ACM, 2016.

[16] "Distribution Test Feeders." https://ewh.ieee.org/soc/pes/dsacom/testfeeders/.

[3] https://pypi.python.org/pypi/PYPOWER

[4] http://pymodbus.readthedocs.io/en/latest/index.html