Development of Cyber-Physical Security Testbed Based on IEC 61850 Architecture

Petr Blazek[1, *], Radek Fujdiak[1,2], Petr Mlynek[1], Jiri Misurec[1]
[1] *Brno University of Technology,*
Technicka 12, CZ-61600 Brno, Czech Republic
[2] *Technical University Ostrava,*
17. Listopadu 2172/15 Ostrava 708 00 Czech Republic
blazekpetr@phd.feec.vutbr.cz

***Abstract*—The paper focuses on security in industrial control systems. Numerous protocols and their incompatibility are undermining the security design. Also, the IEC 61850 standard focuses on these issues. In detail, it deals with the compatibility between protocols and, partly, security. In the context of this work, a testbed together with the traffic generator for IEC 61850 standard and its three main parts – MMS (Manufacturing Message Specification), GOOSE (Generic Object-Oriented Substation Events), and Sampled Values - are designed. Additionally, the used generator is compared with an example of RTU (Remote Terminal Unit) used in standard ICS (Industrial Control Systems) networks. The last part of this work consists of the performance testing of the implemented protocols (MMS, GOOSE, and Sampled Values).**

***Index Terms*—Supervisory control and data acquisition systems; Industrial control systems; Attack; Generator; IEC 61850; Security.**

I. INTRODUCTION

The management of nowadays industrial operations is taken over by operational technologies (OT). The term refers to computing systems, including production line management, mining operations control, oil and gas monitoring, and many others. The major segment within operational technology is comprised of industrial control systems (ICS, often also referred as Industrial Automated Systems - IAS or Industrial Automation and Control System - IACS), which include systems for monitoring and controlling industrial processes, such as oil refinery, power consumption on electricity grids, alarms from building information systems or generally mission-critical applications with a high availability requirement. ICS are divided into two main parts: (i) programmable logic controllers (PLCs) and (ii) discrete control systems (DCS), which also use PLC or some other batch process control device. Moreover, the ICS systems are mostly handled by Supervisory Control and Data Acquisition systems (SCADA), which provide a graphical user interface for operators to observe the system easily, receive possible alarms indicating out-of-band operation or to enter system adjustments to manage the process under control [1], [2].

However, increasingly complex OT and higher interconnections in ICS cause many new opportunities and challenges on different kinds of levels nowadays. The ICS systems are often used in critical industry to control facilities, i.e., hydro-power plants, nuclear power plants, distribution and water treatment facilities, and other facilities with a significant impact on society. These highly interconnected systems are called critical infrastructure (CI) [3] because they have a significant impact on national assets, the basic living needs, and facilities of the population or the public health. An outage of such systems would have a significant impact on the security of the public and national assets. Therefore, one of the essential parts of CI is cybersecurity. Many attacks on ICS systems are based on the already-known attacks from IT networks, such as denial of services, malware, viruses, and others. The threat of attacks on ICS systems can be seen from Kaspersky Lab's report of 2017 [4]. The report identified 322 vulnerabilities in different ICS components for the year 2017. According to a methodology based on the Common Vulnerability Scoring System v3.0, 60 of these vulnerabilities are rated as critical risk, 134 are rated as high risk, 127 are rated as medium risk, and only one is rated as low risk. The networking devices and SCADA devices together contain nearly 50 % of the identified vulnerabilities. This underlines the importance of concentrating on the communication part of the cyber-physical systems involved in OT and ICS. Beyond the identified vulnerabilities, the Kaspersky Lab compiled a list of the most affected areas of the industry, where the highest number of vulnerabilities was found in the CI areas - energy, water industry, and transportation.

Cybersecurity might be approached in different ways. However, two main directions can be identified as follows: (i) security assessment (SA) and (ii) security monitoring (SM). Security assessment includes methods, such as modeling, penetration testing, risk analysis, and others. SA helps to identify the security vulnerabilities of systems or single devices. However, as ICS are critical systems, it is not possible to use the real environment for experiments and tests. Therefore, a secure real-like environment must be used

Manuscript received 24 March, 2019; accepted 2 August, 2019.
This research was supported by The Czech Ministry of Education, Youth and Sports from the Large Infrastructures for Research, Experimental Development and Innovations project "IT4Innovations National Supercomputing Center - LM2015070" and funded by a grant (No. LO1401) from the National Sustainability Program and by a grant (No. VI20172019057) from the Ministry of the Interior of the Czech Republic. For the research, the infrastructure of the SIX Centre was used.

not only to develop mitigations and defense methods, but to test the new devices and technologies or even to train security experts also. The SM method enables the analysis of the communication flow and the discovery of a possible malicious behavior, attacks, and other security incidents via advanced algorithms. To do so, SM needs a sufficiently big dataset to learn communication and behavioral patterns. However, most of the methods are passive, the data is mostly confidential and there is a need for high-quality big data for machine learning purposes, which would make the creation of more accurate and precise analytical and detection algorithms possible [5].

This paper provides recent results from a research project, which deals with the development of a cyber-physical security testbed. The developed testbed provides not only a secure environment for the SA, but also serves as a high-quality data (traffic) generator for SM. The paper contains preliminary results from the implementation of the most adopted International Electrotechnical Commission's (IEC) communication protocol 61850. The rest of the paper is organized as follows. The main contribution of this paper is described in Section II. Section III provides a brief overview of the generators/simulators for protocols from IEC 61850 followed by a vulnerability analysis in Section IV. The general description of the developed testbed is held in Section V with a close description of the IEC 61850 architecture described in Section VI. Finally, Section VII summarizes our conclusion and points out the direction for the future research.

II. RELATED WORK

There are several generators/simulators in SCADA as is the case with IEC 61850. However, most simulators focus only on certain protocols and are unfit to simulate the entire ICS system. The first example [6] is based on RTDS (Real Time Digital Simulator), which is used to test the real-world closed-loop devices. The authors present a simulation of the GOOSE (Generic Object-Oriented Substation Event) and Sampled Values protocols, which are tested on two real relays. However, the simulator serves only for testing of the industrial devices, while it is not possible to simulate the entire network infrastructure. The second work [8] deals with the simulation of GOOSE protocol. The simulator is designed to test in a real network or in a simulated environment, where tens to hundreds of devices can be simulated. The next work [9] deals with the industrial network infrastructure and uses IEC 61850 protocols for the communication among end stations. The work describes the implementation of the GOOSE and Sampled Values protocols, but, in conclusion, the authors assume the implementation of other protocols from this standard in the future work. The authors also state that the simulator can be connected to the real network. The last work [10] describes the simulator based on libiec61850 library, which simulates GOOSE communication. The protocol is implemented in the Riverbed program and simulates the entire infrastructure with real GOOSE communication.

The comparison of the selected generators is shown in Table I. Most of the generators are focusing on software simulation of GOOSE protocol. However, the presented testbed provides an environment for real hardware full implemented IEC 61850 for the security research. Therefore, the main advantage of the presented generator is the full support of IEC 61850, 1 GB/s high-speed link, hardware parts providing the close-to-real environment, monitoring passive interface, and active injection/attack interface. There are few generators, which provide full IEC 61850 stack [10], [11], but these are mostly software simulators without any interface for real traffic injections or attacks simulations. So, they do not fulfill the crucial parameters for security testing.

The main improvement of the state of the art and original contribution of this paper arises from the introduction of our testbed environments, which bring not only one of the new ICS protocol – IEC 61850, but also bring this protocol closer with sufficient information for its implementation in the own environment. Moreover, we introduce several libraries and benchmarks, which should help to set up the hardware setting of the testbed. Last but not least, we also introduce the brief insight into the cybersecurity issue in ICS systems. However, the full security analysis of ICS is behind the scope of this paper and it should serve just as a general overview for threat distribution.

TABLE I. COMPARISON OF DIFFERENT IEC 61850 GENERATORS.

Generator	Ref.	Full IEC 61850	1 GB /s	Hard-ware	Injec-tion	Attacks
Implemented Cyber-physical testbed (*this paper*)	-	✓	✓	✓	✓	✓
Geese: A Traffic Generator for Performance and Security Evaluation	[7]	✗	✓	✗	✗	✗
IEC-61850 GOOSE Traffic modeling and Generation	[9]	✗	✗	✗	✗	✗
Real-time detection of Attacks in IEC 61850	[12]	✗	✓	✓	✓	✓
Microgrid communication design	[13]	✗	✗	✗	✗	✗
Multi-fuction packet generator	[14]	✗	✗	✓	✗	✗
Traffic generator using network emulation	[15]	✗	✗	✗	✗	✗
Real-time Emulation of IEC 61850	[10]	✓	✓	✗	✗	✗
GridSoftware	[11]	✓	✗	✗	✗	✗
Texas Instruments	[16]	✗	✗	✓	✗	✗
GridClone	[17]	✗	✗	✗	✗	✗

III. VULNERABILITY ANALYSIS

Despite the danger that threatens the infected systems CI, many systems are still not sufficiently secured. The proof is also the major attacks that have been carried out on industrial systems. The first major recorded attack was Stuxnet [18], [19], which was discovered by the VirusBlokAda in Belarus in 2010. This worm was designed to reprogram the PCL and hide the changes. Another major attack was called the Night Dragon [20]. The attack aimed at controlling the entire system via advanced tools and techniques, such as password breaks, targeted phishing, abuse of web server vulnerabilities via the SQL injection method, and the security vulnerability in the Windows operating system. Major attacks on CI include Shamoon, which was executed in 2012 against a Saudi Arabian Oil

Company [21]. The attack erased data from more than 35,000 computers in that company. The last known big attack was BlackEnergy [22], [23]. This attack has been refined three times already. At the outset, it was a backdoor Trojan horse that used various components downloaded to the target computer to infect CI. The latest variation has been developed into a complex system that attacks CI in several phases and paralyzes it overall.

Complex attacks usually consist of a series of smaller attacks targeting a specific application or device. These smaller attacks can be divided into known attacks and unknown attacks, which can be further classified according to the attack targets (e.g., network devices, ICS devices). An overview of the most common attacks on ICS devices is provided in Table A-I (in Appendix A). The table shows a description of the attack and its possible detection, and the risk that the attack represents. This analysis was a valuable input for developing the testbed, which must be prepared to simulate the variation of a security incident to provide a close-to-real environment for the machine learning algorithms used, i.e., in monitoring and analytical systems.

IV. TESTBED ENVIRONMENT DESCRIPTION

The testbed environment is displayed in Fig. 1. It contains three main parts:

(i) Red/blue teaming;

(ii) Threat monitoring system;

(iii) Cyber-physical testbed based on IEC 61850.

The (i) is connected via an active interface (allowing each connection and device to enter the network) to achieve high benefits from red/blue team approaches. To achieve high interoperability, the NIST standardization NIST SP 800-161 [24] is considered in the preparation of the environment for security testing. The (ii) is connected via a passive interface (mirroring the main node between the concentrator and the database) not to disturb the communication itself. Moreover, it contains several advanced methods, such as behavioral model analysis, threat detection algorithms, and machine learning parts. Currently, the professional software MENDEL is used, which is a powerful threat detection tool along with SCADA monitoring. A close description of the used methods, as well as of the MENDEL software, might be found in [25]. The (iii) contains RTUs, which are connecting real devices (sensors, meters, relays, and others) and virtualized devices. The communication stack and implementation are described in the next chapter.

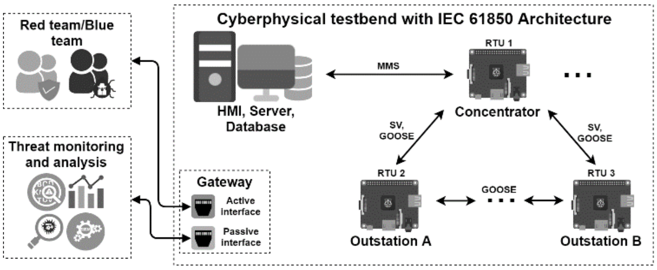

Fig. 1. Testbed environment developed for security testing.

V. CYBER-PHYSICAL TESTBED WITH IEC61850 ARCHITECTURE

The possibility to create the communication and attacks on any network protocols is an essential part of designing the methodology for detecting and filtering attacks. We selected to implement the widely used IEC 61850 standard. This standard includes several protocols to guarantee a certain quality of cyber security/safety in SCADA communication.

A. Description of IEC 61850

IEC 61850 defines the standardized methods for building communication networks and integration of devices in industrial systems. The primary goal is to enable simple device communication from different manufacturers. This standard collects comprised overall 10 documents. To ensure reliable communication between all devices in the system, communication protocols are defined on all layers of the ISO/OSI model. For example, the transmission speed is crucial for critical data. For this reason, critical data from the Application layer is routed directly to the Data Link Layer using the GOOSE protocol.

The most important parts of this standard include three basic protocols. The first one is the Manufacturing Message Specification (MMS), which uses messaging systems for transferring real-time data and control information between devices. This is an application protocol that communicates over transport to physical layers. Another essential protocol is called GOOSE. These events are used for the fast transferring of critical data over the entire system. One important thing is that the response delay must not be higher than 4 ms. Fast data transfer is used for communication only on the Link Layer. The third protocol Sampled Values (SV) is very similar to GOOSE, but it is not used for a critical event. This protocol sends high-speed multi-cast messages that contain user-defined values. The second layer (Link Layer) is used for the communication of the ISO/OSI model, same as with GOOSE. There are several possibilities to implement the considered traffic generator. It is possible to simulate traffic in a simulation tool (software based) or directly implement protocols into devices (hardware based). For our implementation, we selected the second variant with library libiec618501 to obtain the environment, which is the closest to the real network. Our main idea was to implement a system that could contain simulated network elements as well as real devices communicating with IEC 61850. The library libiec61850 provides an implementation of all three mentioned protocols (MMS, GOOSE, and SV). We opted for the well-known single-board computer Raspberry Pi 3B+ with the operating system Raspbian as our main hardware platform. This platform can be extended by several communication modules (LTE modem, RS232, RS485, and others) that are used in ICS systems. Another advantage is that the general purpose inputs/outputs (GPIO) might be used for connecting various devices from ICS (relays, sensors, and others).

B. Generator Structure

The main idea behind the design was to simulate a real-world device from ICS networks. The Remote Terminal Unit (RTU) was decided to be used as a template. All three of the above-mentioned IEC 61850 protocols (MMS, GOOSE, and Sampled Values) are used in RTU for the communication among devices or between the device and HMI [26]. An example of RTU is shown in Fig. 2. It is a basic RTU for a transformer station that contains two relays and six measuring elements. Furthermore, RTU communicates with GPRS/LTE with a control and processing unit, where data from ICS protocols are processed.

Based on RTU, a generator was created. Its structure is displayed in Fig. 1. The main parts of the generator include three Raspberry Pi and one standard desktop server. First RPi labeled as Concentrator is the equivalent of the RTU from Fig. 2. The Concentrator is used to collect data from the Outstation A and Outstation B devices that are sent via MMS to the HMI. Therefore, the Concentrator performs the function of a server in the client-server communication. RPi station labeled as Outstations is used as a simulated ICS device or as a provider for devices that do not directly communicate with the IEC 61850 standard (sensors, relays, and others.). The last part is a desktop computer (HMI), which is equivalent to the SCADA control and processing block from Fig. 1. The connection between the stations is connected via an Ethernet cable and a standard switch.

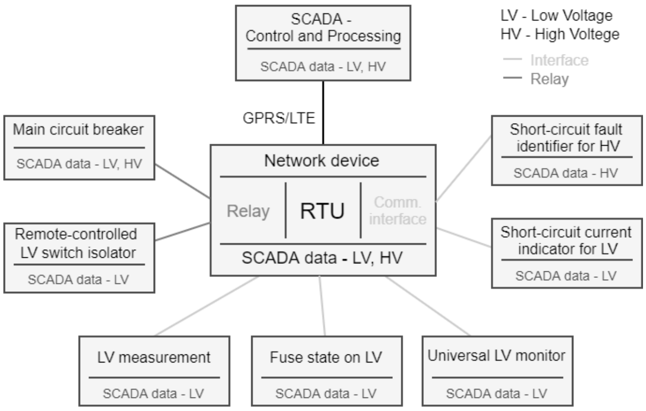

Fig. 2. Example of basic SCADA RTU.

1) *MMS implementation*

The MMS protocol is based on the client-server communication. In libiec61850, there are two libraries *iec-61850server.h* and *iec-61850client.h* that provide client-server communication. The MMS protocol is implemented between the Concentrator and the HMI as shown in Fig. 1. During the usual network traffic, one MMS request per second is generated., the Concentrator responds with a message that contains the GPIO data of all RTU (including own) to this request. The size of generated messages is 88 bytes per request and 125 bytes per response.

2) *GOOSE implementation*

Compared to MMS, GOOSE is based on the multi-cast communications called publisher-subscriber. The communication is mediated through three libraries, namely *goosepublisher.h*, *goosesubscriber.h*, and *goosereceiver.h*. The publisher sends multicast messages that are received by the subscribers based on an identifier. The library *goosereceiver.h* is an additional library for subscribers and it is used to receive GOOSE messages. As shown in Fig. 1, the GOOSE messages are generated from all RTU stations. For the Outstation A and B, each message contained a total amount of 195 bytes. The generated GOOSE messages from Concentration Station were larger because they transmit data for both slave stations in one message. The total amount of one message was 422 bytes.

3) *SV implementation*

As GOOSE, Sampled Values use the publisher-subscriber messaging architecture. This protocol is used to send periodic data messages (e.g., values from an electrometer). In libiec61850, there are two main libraries called *svpublisher.h* and *svsubscriber.h*, which are used to mediate the communication. As shown in Fig. 1, SV messages are generated only from the Outstation A and Outstation B. The receiver of 203-byte messages was the Concentrator, who passed on the data to HMI via the MMS protocol.

C. *Performance Testing of the Testbed*

In the previous sections, a test testbed based on RPi was introduced. This section describes performance testing of all RPi. The CPU utilization and transmission speed (incoming and outgoing) were monitored during the tests. The testing was divided into three parts according to the protocols used (MMS, GOOSE, and SV). During the tests, the generated traffic consisted of one protocol between the stations that support it. Each test lasted fifteen minutes and was repeated five times for each protocol. The size of the generated messages corresponded to the values given in the section B (Generator structure).

Table II shows the average values for the CPU utilization and transmission speed of all the tests performed. In addition to the CPU utilization and transmission speed, the average of the maximum number of packets generated per second is shown in the table below.

TABLE II. AVERAGE VALUES OF ALL TESTS PERFORMED.

Protocol	**Station**	**CPU**	**Transmission speed**	
		[%]	**[Mbit/s]**	**[packets/s]**
MMS	Concentrator	80,91	159,24	87 236,26
GOOSE	Concentrator	14,98	96,18	19 972,29
	Outstation A	82,8	184,72	122 072,23
	Outstation B	88,98	185,72	122 112,45
Sampled Values	Concentrator	0,77	-	-
	Outstation A	79,45	206,42	129 889,42
	Outstation B	85,69	208,33	130 456,02

1) *Performance testing of MMS protocol*

The number of packets per second was increased from normal traffic, when one packet per second is generated, to an average of 87 236,26 packets per second (request and response). Figure 3 displays the CPU utilization of one test, which was around 80 percent for one test~~, which~~. This value corresponds to the value in Table II and the remaining tests, which were very similar. The average transmission speed was almost 160 Mbit/s.

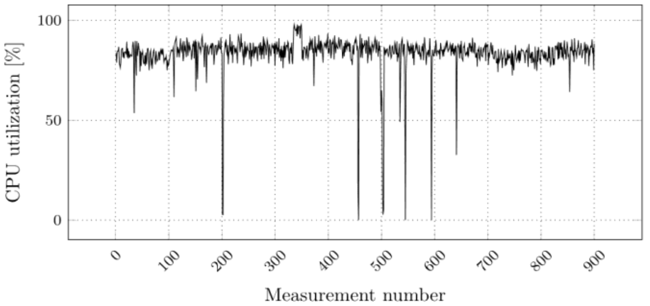

Fig. 3. Example of one measurement for CPU monitoring during MMS generation.

2) *Performance testing of GOOSE protocol*

The second testing was focused on GOOSE protocol, which communicates on the second layer of the OSI/IOS model using multicast frames. As mentioned in Section B

(Generator structure), messages were generated from all RTUs. Table II shows that the average utilization and transmission speed for both Outstations was very similar. Both Outstations reached more than 180 Mbit/s with the utilization of more than 80 percent. For the Concentrator, approximately half of the transmission speed was reached, but the processor utilization was only 15 percent. This fact was caused by overloading the network card itself, when it was unable to process a large number of GOOSE messages from the remaining stations and generate its messages. An example of one of the GOOSE testing is provided in Fig. 4, which shows the utilization of all used RTUs. The remaining tests had a very similar pattern.

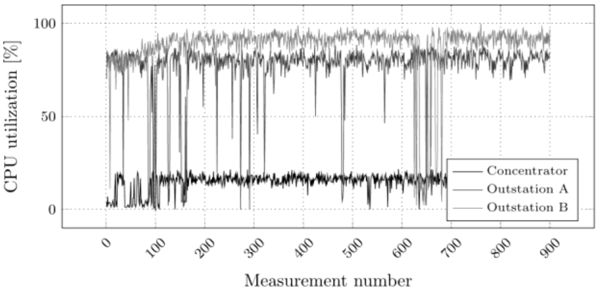

Fig. 4. Example of one measurement for CPU monitoring during GOOSE generation.

3) *Performance testing of SV protocol*

The last test was focused on the Sampled Values protocol, which is very similar to GOOSE. During the test, messages from RTU 2 and RTU 3 were sent to the Concentrator station. When generating Sampled Values messages, the highest transmission speed (over 200 Mbit/s) was achieved with approximately the same processor utilization on the Outstations as with GOOSE testing. Interesting is the CPU utilization of messages receiver because, during the tests, there was almost no use of the processor as seen in Fig. 5.

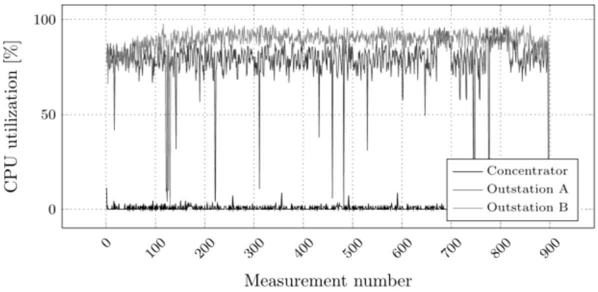

Fig. 5. Example of one measurement for CPU monitoring during Sampled Values generation.

The above tests summarize the traffic limits for three major IEC 61850 protocols (MMS, GOOSE, and SV) implemented on Raspberry Pi single-board computers. Whiting a standard RTU unit, the data traffic is at a maximum of tens of Mbit per second. According to the results of Table II, RPI stations are more than sufficient to simulate the RTU station operation. The testing also proves that, even if the testbed is expanded with additional devices (Concentrator, Outstation, and Real device), the RPi unit will be powerful enough to transmit all the traffic.

VI. CONCLUSIONS

Security in ICS networks is a frequently debated, but still highly underestimated topic. Many protocols and their incompatibility undermine the security design. Therefore, an extensive analysis of the most common vulnerabilities of ICS protocols was performed, together with giving clear hints for mitigation and detection. Further, one of the most promising ICS protocols IEC 61850 was introduced. The analysis of current solutions was presented, and the common imperfections of these solutions identified. Among these solutions, we bring a high-speed laboratory environment with hardware emulators, which fills the identified gaps by implementing the main parts of IEC 61850 stack and by giving a possibility of injecting, attacking or capturing the communication. Further research should be focused on the implementation of the security incident scenarios and on extending the communication stack for the synchronization part of IEC 61850 (SNTP).

APPENDIX A – ICS PROTOCOL VULNERABILITY DESCRIPTION

TABLE A-I. LIST OF KNOWN AND UNKNOWN ATTACKS [4], [27]–[31].

Attack	Risk	Process	Detection
Known types of attacks			
Network Mapping	Identifying possible targets for further attacks.	Scanning services within a network segment or multiple services within a single device.	Signature detection or anomaly detection.
Firmware detection	Identifying a specific version to which a particular type of attack can be executed.	System version query.	Signature detection or anomaly detection.
Configuration Error	Access control to device or application resources (data, configuration information, user data).	It occurs for each type of communication differently or as a specific character list.	Known - signature detection. Unknown - anomaly detection (Difficult to detect).
Application Error	Code injection – admin, access, data steal or denial of service.	It occurs for each type of communication differently or as a specific character list.	Known - signature detection. Unknown - Difficult to detect.
(D) DoS	Denial of service availability for users.	Increased communication focused on resource depletion.	Communication analysis or signature detection.
Unknown types of Attacks			
Changing of Database Configuration	Shutdown devices that are controlled by the configuration database.	Different for each type of comm., not recognizable from the normal behavior.	Unauthorized access, or abnormality of the given comm. in terms of time distribution.
Changing Parameters	A change in the behavior of the servicing device.	Changing in communication or sending unexpected variables.	Unauthorized access, behavior analysis of administrator, time analysis detection
Zero-Day Attacks on Application/Configuration	Unauthorized access to resources.	The attack is based on the attacked application/configuration.	Attack-related activities, such as anomalous data transfers, CnC machine communication, abnormal behavior.

REFERENCES

[1] W. Graham, "OT, ICS, SCADA – What's the difference?", *Kuppingercole Analysts*, 2015. [Online]. Available: https://www.kuppingercole.com/blog/willia-mson/ot-ics-scada-whats-the-difference

[2] J. M. Gutierrez-Guerrero and J. A. Holgado-Terriza, "iMMAS an industrial meta-model for automation system using OPC UA", *Elektronika ir Elektrotechnika*, vol. 23, no. 3, pp. 3–11, 2017. DOI: 10.5755/j01.eie.23.3.18324.

[3] S. M. Rinaldi, J. P. Peerenboom, and T. K. Kelly, "Identifying, understanding, and analyzing critical infrastructure interdependencies", *IEEE Control Systems*, vol. 21, no. 6, pp. 11–25, 2001. DOI: 10.1109/37.969131.

[4] Kaspersky, *Threat Landscape for Industrial Automation Systems in H2 2017*, 2018. [Online]. Available: https://ics-cert.kaspersky.com/reports/2018/03/26/thre-at-landscape-for-industrial-automation-systems-in-h2-2017/

[5] S. Ekins, J. S. Freundlich, and R. C. Reynolds, "Are bigger data sets better for machine learning? Fusing single-point and dual-event dose response data for Mycobacterium tuberculosis", *Journal of chemical information and modeling*, vol. 54, no. 7, pp. 2157–2165, 2014. DOI: 10.1021/ci500264r.

[6] R. Kuffel, D. Ouellette, and P. Forsyth, "Real time simulation and testing using IEC 61850", in *Modern Electric Power Systems (MEPS), Proc. of the International Symposium*, 2010.

[7] Y. Lopes, D. C. Muchaluat-Saade, N. C. Fernandes and M. Z.Fortes, "Geese: A traffic generator for performance and security evaluation of IEC 61850 networks", in *Proc. of 2015 IEEE 24th International Symposium on Industrial Electronics (ISIE)*, 2015. DOI: 10.1109/ISIE.2015.7281552.

[8] B. A. Souza, N. S. D. Brito, E. C. Gurjão, J. A. Sa, R. R. R. Ribeiro, M. T. Barreto, and U. A. Carmo, "An IEC 61850 network simulator", in *Proc. of 2010 IEEE/PES Transmission and Distribution Conference and Exposition: Latin America (T&D-LA)*, 2010. DOI: 10.1109/TDC-LA.2010.5762930.

[9] O. Hegazi, E. Hammad, A. Farraj, and D. Kundur, "IEC-61850 GOOSE traffic modeling and generation", in *Proc. of 2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2017. DOI: 10.1109/GlobalSIP.2017.8309131.

[10] S.-H. Hwang, "GOOSE traffic generator using network emulation", *The Journal of The Institute of Internet, Broadcasting and Communication*, vol. 16, pp. 209–214, 2016. DOI: 10.7236/JIIBC.2016.16.1.209.

[11] *Grid Software Products: A better engineering our software can help enable (Smarter. Faster. Better.)*, 2018. [Online]. Available: http://www.gridsoftware.com/products.html

[12] L. E. da Silva and D. V. Coury, "A new methodology for real-time detection of attacks in IEC 61850-based systems", *Electric Power Systems Research*, vol. 143, pp. 825–833, 2017. DOI: 10.1016/j.epsr.2016.08.022.

[13] I. Ali and S. M. S. Hussain, "Communication design for energy management automation in microgrid", *IEEE Transactions on Smart Grid*, vol. 9, no. 3, pp. 2055–2064, 2018. DOI: 10.1109/TSG.2016.2606131.

[14] W. Wei, H.-b. Li, and H.-m. Cheng, "A multi-function IEC 61850 packet generator based on FPGA", *Measurement Science and Technology*, vol. 27, no. 7, p. 075901, 2016. DOI:10.1088/0957-0233/27/7/075901.

[15] S.-H. Hwang, Y.-S. Im, H.-Ch. Song, and J.-D. Park, "Real time emulation of IEC61850 SV, GOOSE and MMS using NS-3", *Journal of Engineering and Applied Sciences*, vol. 13, no. 3, pp. 634–638, 2018. DOI: 10.3923/jeasci.2018.634.638.

[16] *Packet Processing Engine Reference Design for IEC61850 GOOSE Forwarding*, 2016. [Online]. Available: http://www.ti.com/lit/ug/tidubo1/tidubo1.pdf

[17] G. Clone, *Grid Clone Products*, 2016. [Online]. Available: http://www.gridclone.com/p/home

[18] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon", *IEEE Security & Privacy*, vol. 9, no. 3, pp. 49–51, 2011. DOI: 10.1109/MSP.2011.67.

[19] T. M. Chen and S. Abu-Nimeh, "Lessons from Stuxnet", *Computer*, vol. 44, no. 4, pp. 91–93, 2011. DOI: 10.1109/MC.2011.115.

[20] *Global Energy Cyberattacks: "Night Dragon*", 2011. [Online]. Available: https://securingtomorrow.mcafee.com/wp-content/uploads/2011/02/McAfee_NightDragon_wp_draft_to_customersv1-1.pdf

[21] Z. Dehlawi and N. Abokhodair, "Saudi Arabia's response to cyber conflict", in *Proc. of 2013 IEEE International Conference on Intelligence and Security Informatics*, 2013. DOI: 10.1109/ISI.2013.6578789.

[22] D. U. Case, "Analysis of the cyber attack on the Ukrainian power grid", *Electricity Information Sharing and Analysis Center (E-ISAC)*, 2016. [Online]. Available: https://ics.sans.org/media/E-ISAC_SANS_Ukraine_DUC_5.pdf

[23] S. Raj and B. Christiaan, *Updated BlackEnergy Trojan Grows More Powerful*, 2016.

[24] J. Boyens, C. Paulsen, R. Moorthy, N. Bartol, and S. A. Shankles, "Supply chain risk management practices for federal information systems and organizations", *NIST Special Publication*, vol. 800, p. 1, 2014. DOI: 10.6028/NIST.SP.800-161.

[25] *MENDEL: Security for professionals*, 2018. [Online]. Available: https://www.greycortex.com/mendel

[26] S. Sanjay, "Substation Communication with IEC 61850 and Application Examples", ABB, 2016. Available. [Online]: http://www04.abb.com/global/seitp/seitp202.nsf/0/4d1c836b9e7fdb67c12580870047d7c8/$file/1.Chile_+ABB+_Substatio+communication+with+IEC+61850+and+application+examples.pdf

[27] Z. Drias, A. Serhrouchni, and O. Vogel, "Taxonomy of attacks on industrial control protocols", in *Proc. of 2015 International Conference on Protocol Engineering (ICPE) and International Conference on New Technologies of Distributed Systems (NTDS)*, 2015. DOI: 10.1109/NOTERE.2015.7293513.

[28] G. Andrew, "The Top 20 Cyber Attacks Against Industrial Control Systems", Waterfall Security Solutions, 2017.

[29] I. Zolotova, R. Hosak, and M. Pavlik, "Supervisory control sustainability of technological processes after the network failure", *Elektronika ir Elektrotechnika*, vol. 18, no. 9, pp. 3–6, 2012. DOI: ttp://dx.doi.org/10.5755/j01.eee.18.9.2794.

[30] J. Mocnej, W. K. Seah, A. Pekar, and I. Zolotova, "Decentralised IoT architecture for efficient resources utilisation", *IFAC-PapersOnLine*, vol. 51, no. 6, pp. 168–173, 2018. DOI: 10.1016/j.ifacol.2018.07.148.

[31] J. Mocnej, T. Lojka, and I. Zolotova, "Using information entropy in smart sensors for decentralized data acquisition architecture", *in Proc. of 2016 IEEE 14th International Symposium on Applied Machine Intelligence and Informatics (SAMI)*, 2016, pp. 47–50. DOI: 10.1109/SAMI.2016.7422980.