Proceedings Paper:
Kemmeugne, A, Jahromi, AA, Kundur, D et al. (1 more author) (2021) Towards cyber-resilient telecontrol commands using software-defined networking. In: MSCPES '21: Proceedings of the 9th Workshop on Modeling and Simulation of Cyber-Physical Energy Systems. CPS-IoT Week '21: Cyber-Physical Systems and Internet of Things, 19-21 May 2021 ACM . ISBN 978-1-4503-8608-1

https://doi.org/10.1145/3470481.3472707

Towards Cyber-Resilient Telecontrol Commands Using Software-Defined Networking

Anthony Kemmeugne
anthony.kemmeugne@mail.utoronto.ca
University of Toronto
Toronto, Ontario, Canada

Amir Abiri Jahromi
University of Leeds
Leeds, United Kingdom
A.AbiriJahromi@leeds.ac.uk

Deepa Kundur
Univerisity of Toronto
Toronto, Canada

Marthe Kassouf
Institut de Recherche d'Hydro QuÃľbec
Varennes, QuÃľbec, Canada

ABSTRACT

Cybersecurity enhancement of power systems has become one of the main objectives of utility managers and regulatory agencies because of the increasing number of cyberattacks against critical infrastructures. In this paper, we investigate the application of software-defined networking for improving the cyber-resilience of power systems in the presence of cyberattacks using false telecontrol commands. It is first demonstrated that cyberattackers can use false telecontrol commands to separate a power plant from a power grid or trip a major transmission line. Next, it is shown that software-defined networking can significantly enhance the cyber-resilience of power systems in the presence of cyberattacks using false telecontrol commands compared to legacy communication networks. This is because the source, destination and protocol of telecontrol commands can be examined and verified in software-defined networking before communication packet forwarding actions take place. Moreover, primary and back-up routes of telecontrol commands can be pre-engineered in software-defined networking to counteract potential cyberattacks.

CCS CONCEPTS

• **Networks** → **Cyber-physical networks**; • **Security and privacy** → *Malware and its mitigation*; Distributed systems security; • **Computing methodologies** → Modeling and simulation.

KEYWORDS

Cyberphysical systems, telecontrol commands, software-defined networking, cyber-resilience, co-simulation

ACM Reference Format:
Anthony Kemmeugne, Amir Abiri Jahromi, Deepa Kundur, and Marthe Kassouf. 2021. Towards Cyber-Resilient Telecontrol Commands Using Software-Defined Networking. In *9th Workshop on Modeling and Simulation of Cyber-Physical Energy Systems (MCPES'21), May 19–21, 2021, Nashville, TN, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3470481.3472707

MCPES'21, May 19–21, 2021, Nashville, TN, USA

ACM ISBN 978-1-4503-8608-1/21/05…$15.00
https://doi.org/10.1145/3470481.3472707

1 INTRODUCTION

Supervisory control and data acquisition (SCADA) systems provide operators ability to monitor and control power systems [1]. Legacy SCADA systems rely on private communication networks without remote access which minimizes the accessibility of cyber intruders and diminishes potential cyber threats. Yet, the introduction of standard and interoperable communication protocols like IEC 61850, adoption of Internet of Things devices and cloud services as well as connection of SCADA systems to corporate services and enterprise networks are expected to compromise the cybersecurity of SCADA systems [2, 3]. Thus, there is a pressing need for novel security measures to enhance the cybersecurity of SCADA systems.

Cybersecurity of SCADA systems has received considerable attention over the past decade in particular after the cyberattack against a waste management system in Queensland [4] and cyberattacks against the Ukrainian power grid in 2015 and 2016 [5], [6]. A comprehensive literature survey of research works on SCADA cybersecurity has been provided in [7]. A framework for cyber vulnerability assessment of SCADA in power systems has been proposed in [8, 9]. In [10], power system reliability has been evaluated considering cyberattacks against SCADA systems.

Telecontrol is a critical function in SCADA systems which can be exploited by cyberattackers to execute a wide range of attacks against power systems including harmful network topology changes, critical asset tripping or unwanted load shedding [11–13]. Cyberattacks using false telecontrol commands are detrimental to power systems for several reasons. First, coordinated cyberattacks using false telecontrol commands can cause significant impact in a short period of time. Second, the sluggish nature of data collection by SCADA systems prevents any potential corrective actions to thwart damages caused by cyberattacks using false telecontrol commands. This is because operators may become aware of the consequences of such attacks only after damage occurrence [12]. Additionaly, false data injection attacks can be used by cyberattackers against SCADA data collection in harmony with false telecontrol command injection attacks to hamper corrective actions by operators. Lastly, detection and mitigation of cyberattacks which employs false telecontrol commands is difficult considering the existing systems and practices since telecontrol commands lack authentication or other security features [12].

The previous research works for enhancing the cybersecurity of telecontrol commands have been focused on distributed security solutions and intrusion detection systems. A semantic analysis

framework based on a distributed network of intrusion detection systems has been proposed in [11] to detect cyberattacks using false telecontrol commands. This framework has been extended in [12, 13] by considering a power flow analysis capable of estimating the execution consequences of telecontrol commands. A distributed security solution has been proposed in [14] which performs real-time power system analysis to evaluate the impact of the control commands before execution. The semantics of Generic Object Oriented Substation Events (GOOSE) messages have been employed in [15] to identify abnormal behaviors of GOOSE telecontrol commands in the IEC 61850 compliant systems. This is while little or no attention has been given to cyber-resilience of communication networks involved in telecontrol functions.

As communication networks in substations move away from copper wires toward Ethernet-based connections, it is crucial to investigate the cyber-resilience of available Ethernet-based communication network technologies due to stringent requirements of automation and control functions like telecontrol functions in substations. The term *Resilience* is defined by the National Academics as "the ability to prepare and plan for, absorb, recover from, and more successfully adapt to adverse events" [16]. In this paper, we use the term cyber-resilience to refer to the ability of the communication network to plan for and mitigate cyberattacks against telecontrol commands. Software-defined networking is a programmable communication technology that has received considerable attention over the past few years for satisfying the scalability, reliability, and ease of implementation demanded by automation and control functions for power systems. D. Jin et al. investigates the self-healing capabilities of SDN compared to legacy network in the context of microgrid operation [17]. In [18], the authors investigated the security of SDN infrastructure when applied to smart grids. In this paper, we use the term cyber-resilience to refer to the ability of the communication network to withstand and mitigate cyberattacks against telecontrol commands.

The main contributions of this paper are as follows.

- The development of a co-simulation-based platform for modeling and implementation of SDN technologies in power grid operational environments.
- The use of this platform for cyberattack simulation against telecontrol commands and for the assessment of SDN cyber-resilience.

The remainder of this paper is organized as follows. Section II presents the attack model. The basics of software-defined networking are briefly discussed in Section III. A co-simulation environment for cybersecurity analysis of telecontrol functions is described in Section IV. Lastly, simulation results are provided in Section V before providing concluding remarks in Section VI.

2 ATTACK MODEL

In this paper, we assume that cyberattackers have remote access to the wide area network connecting a control center and substations as illustrated in Fig. 1. Therefore, they are able to sniff and inject telecontrol commands. The attackers are assumed to acquire the required remote access either by using legitimate stolen operators credentials or by recruiting an employee with legitimate access to the network and with the capability to connect a malicious device for attack execution.

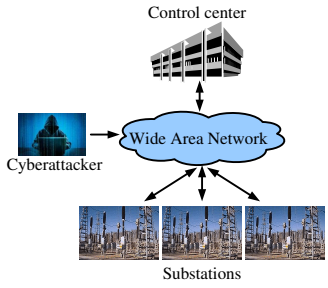

Figure 1: Cyberattacks using telecontrol commands.

After securing the initial network access, cyberattakers can take advantage of the absence of encryption in SCADA communication protocols to sniff packets containing substation information such as measurements, telecontrol commands, and status of circuit breakers. Moreover, they can obtain critical information about the network topology and the substation critical assets to perform a high impact attack.

In this work we consider that cyberattackers execute a False Data Injection attack against telecontrol commands. In that scenario, the attackers gain access to the wide area network and actively inject false telecontrol commands toward substations to achieve their objectives.

3 SOFTWARE-DEFINED NETWORKING

Software-defined networking is a paradigm shift in communication network management which decouples control plane from data plane [19]. In legacy communication networks, data plane and control plane are both integrated into network devices. As such, network devices determine paths of communication packets using distributed algorithms and protocols such as rapid spanning tree protocol (RSTP). In contrast, paths of communication packets in software-defined networking are centrally configured by applications in SDN application plane through SDN controller. One of the benefits of this paradigm shift is that primary and back-up paths for every communication packet can be pre-engineered considering the source, destination and protocol of communication packets.

Communication networks in software-defined networking consist of three planes including application plane, control plane and data plane as illustrated in Fig. 2 [20]. The application plane is responsible for decision making about network resources/functions, and communicates with the control plane using northbound application programming interface (API). The control plane manages the network devices in the data plane based on instructions received from the application plane. The control plane communicates with the data plane using southbound API. The OpenFlow protocol is used by SDN controller to enforce forwarding rules to network devices and collecting network statistics upon the application plane requests.

OpenFlow employs three functions including matches, actions and counters to control and monitor the data plane. When a communication packet enters an OpenFlow-enabled switch, the switch

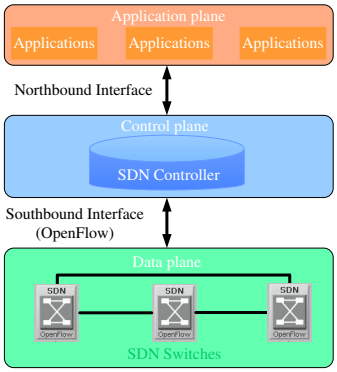

Figure 2: Software-defined networking layers.

examines the header of the packet against match fields and based on rule matches performs appropriate actions. Counters are used in OpenFlow-enabled switches to maintain statistics on communication traffic. The match-action pair in software-defined networking provides the capability to examine and verify different fields in communication packets such as source, destination and protocol before forwarding actions take place. This provides much more advanced security control features compared to legacy communication networks. It is worth noting that the implementation of defence strategies like MAC address filtering in legacy communication networks is an error-prone and arduous task because it should be implemented manually for every single switch in the communication network.

Network devices can be configured reactively or proactively in software-defined networking. In the proactive case, network devices are configured in advance by SDN controller and communication packets that do not match the rules in flow tables of network devices are dropped. In contrast, SDN controller operates in reaction to new communication traffic in the reactive case. In the reactive case, communication packets that do not match the rules in flow tables of network devices are sent to SDN controller. Network applications then configure new rules for new packets in network devices through SDN controller. The proactive approach eliminates unnecessary delays introduced by communication between network devices and SDN controller to manage new communication packets and therefore is more suitable for applications in power systems. Moreover, proactive approach is advantageous from cybersecurity perspective since SDN controller can be removed after pre-engineering the primary and back-up routes of communication packets. Therefore, the single point of failure vulnerability of SDN controller can be eliminated in the proactive case. SDN was originally designed to operate in reactive mode of operation since it enables on-demand resource allocation and self adaptive configuration. In general OT practices lean toward the use of static system configurations that require less updates and can work in stand alone mode for a long period unlike IT practices where updates and interactions with the system operators are more frequent.

SDN switches operate based on a deny-by-default model in contrast to legacy switches that operate based on plug-and-play model [21]. As such, communication packets that do not match flow tables in SDN switches are dropped by default. Unmatched packets further can be forwarded to intrusion detection systems for further investigation. Moreover, SDN switches operate based on whitelisting of communication packets. This is in contrast with legacy communication networks that operate based on blacklisting of communication packets. An important benefit of whitelisting compared to blacklisting is that legitimate communication packets entering a wrong port will be dropped. Therefore, cybersecurity controls can be implemented to drop all communication packets that do not match the pre-engineered traffic, ports and devices.

Testing and validation of cyber-resilience solutions like software-defined networking for power systems is a difficult task due to the complex interactions between cyber and physical elements as well as unforeseen challenges that may occur during integration of these solutions [22]. Pilot testbeds and emprical prototyping are commonly used for this purpose. As such, we employ a co-simulator based on OPAL-RT simulator and Riverbed Modeler to replicate the cyber-physical interactions in a SDN-enabled power system.

4 A CO-SIMULATOR FOR CYBERSECURITY ANALYSIS OF TELECONTROL COMMANDS

Co-simulation has received considerable attention in recent years in particular for cyber-physical security analysis of power systems. This is mainly because of the rapid integration of communication and software components with physical devices in power systems [23]. The existing co-simulators in the literature for conducting cyber-physical security analysis can be classified into offline and real-time co-simulators. The main advantages of real-time co-simulators are the ease of time synchronization, integration of hardware in the co-simulation environment and scalability.

In this paper, we employ a real-time co-simulator based on OPAL-RT simulator and Riverbed Modeler to investigate cyberattacks against power systems using false telecontrol commands. Moreover, the co-simulator is employed to demonstrate the benefits of software-defined networking for improving the cyber-resilience of power systems compared to legacy communication networks.

OPAL-RT simulator is a real-time simulator which provides the capability to simulate power systems. Moreover, we use OPAL-RT to investigate the impact of a cyberattack against telecontrol commands on power systems. This simulator interfaces with other simulators or hardware through its input/output (I/O) modules and Ethernet ports. Riverbed Modeler is a flexible communication network simulator that models a variety of protocols, technologies and network types and provides real-time simulation capabilities through system-in-the-loop (SITL) feature. As illustrated in Fig. 3, OPAL-RT simulator and Riverbed Modeler communicate using IEC 61850 data packets through network interface cards.

The data plane of software-defined networking is implemented in Riverbed Modeler. The SDN controller is implemented using OpenDayLight. It runs as a service in a Virtual Machine powered by Oracle VM. The virtual machine connects to Riverbed Modeler through an SITL "SDN Controller" illustrated Fig3 . The application plane is implemented using Postman API. Postman is an application that interact with HTTP API. In our case it communicates with

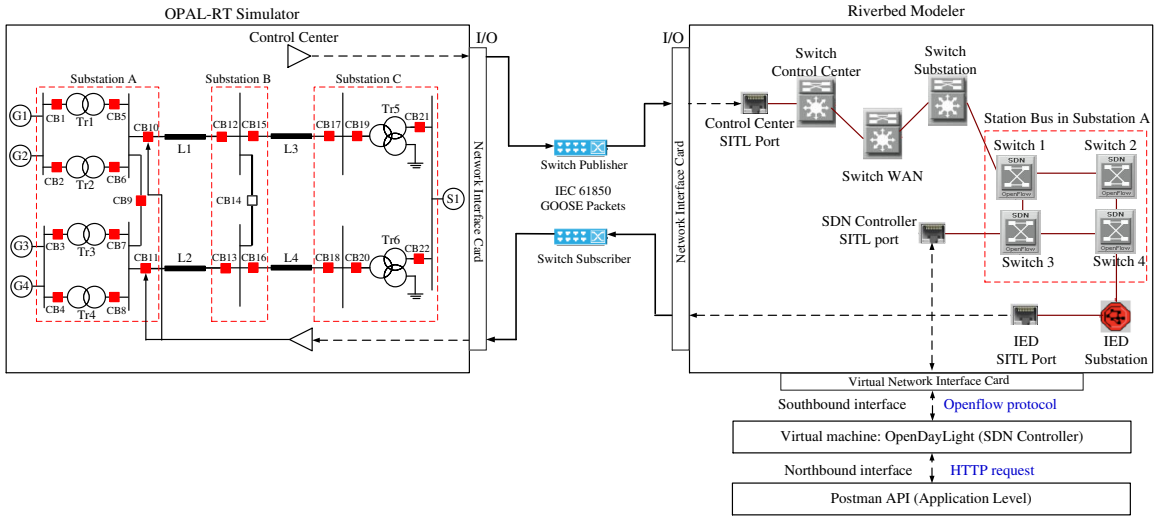

Figure 3: Schematic representation of the co-simulator based on OPAL-RT simulator, Riverbed Modeler and Opendaylight SDN controller.

OpenDayLight using HTTP protocol and OpenDayLight communicates with the data plane in Riverbed Modeler using OpenFlow protocol. Telecontrol command routes for our studies are programmed in Postman and enforced to SDN switches in the data plane using SDN controller in OpenDayLight. It is worth noting that Riverbed modeler, Postman, and Opendaylight virtual machine are all running on a single machine, and only OPAL-RT runs on an external machine.

5 SIMULATION RESULTS

Fig. 4 illustrates the IEEE power system relaying committee (PSRC) D6 benchmark test system [24]. The test system connects a power plant with four generators to a 230 kV transmission network through two parallel 500 kV transmission lines. The 230 kV transmission system is modeled by an ideal voltage source. It is assumed that the objective of cyberattackers is to disconnect the power plant from the transmission system by tripping the circuit breakers CB10 and CB11 through false telecontrol commands.

Two case studies are conducted here. In the first case study, the station bus communication network of substation A is implemented using legacy communication networks. In the second case study, the station bus communication network of substation A is implemented using software-defined networking. The assumption in the case studies is that cyberattackers have access to the wide area network between the control center and substation A.The standard IEC61850-90-12 [25] states that WAN communication can be achieved using routable GOOSE (R-GOOSE) based on UDP protocol, IP tunneling which encapsulate GOOSE packets in an IP header and lastly layer-2 GOOSE packets. In the case of layer 2 packets being sent over the WAN, VLAN tagging is used to separate the local traffic (within a substation) and WAN traffic (between substations/control center). In this paper, we use GOOSE messages for telecontrol commands in the case studies. Therefore, layer 2 switches are used to implement the

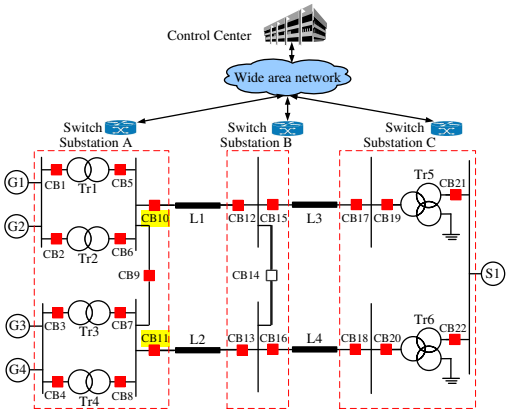

Figure 4: IEEE PSRC D6 Test system.

WAN. Nevertheless, the same approach can be used for telecontrol commands based on R-GOOSE without loss of generality.

5.1 Case Study I: False Telecontrol Command Injection Attacks Against Power Systems – Legacy Communication Networks

In this case study, the station bus in substation A is implemented in Riverbed Modeler based on legacy communication networks as illustrated in Fig. 5. A tool named tcpreplay is employed in combination with wireshark to inject false telecontrol commands to the wide area network in Riverbed Modeler. Wireshark tool is an open source software which is able to monitor and save communication packets. First, the OPAL-RT simulator is employed to generate GOOSE packets containing false telecontrol commands. Wireshark is then employed to save the false telecontrol packets. In the case studies, the destination MAC address of false telecontrol commands

match the MAC address of circuit breakers CB10 and CB11. Moreover, GOOSE packets are used to send false telecontrol commands. Yet, the source MAC address of false telecontrol commands is considered to be different from the MAC address of the control center. It is worth noting that other scenarios such as sending telecontrol commands with wrong destination address or wrong communication protocol as well as telecontrol commands from wrong ports can be tested similarly using the co-simulator presented in this paper. Nevertheless, these scenarios are not discussed in the paper for the sake of brevity.

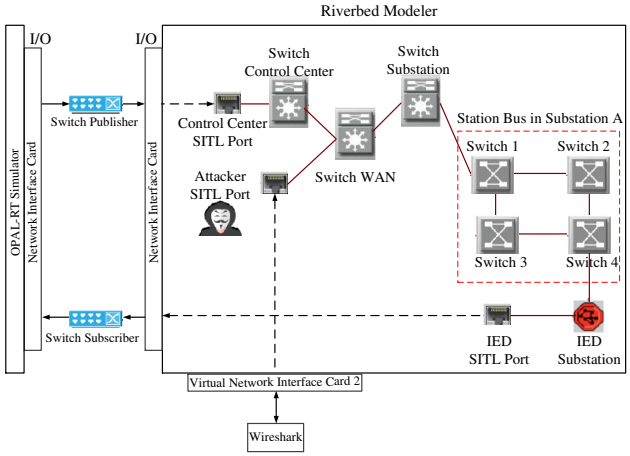

Figure 5: Communication network between the control center and IED in the substation A - station bus based on legacy communication networks.

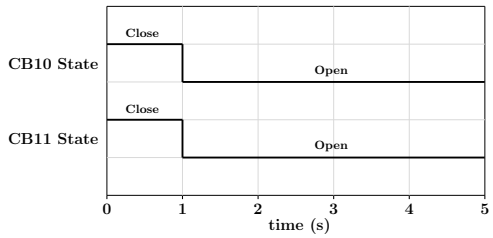

Figure 6: Status of circuit breakers CB10 and CB11 in Case Study I.

The IEEE PSRC D6 test system is co-simulated using OPAL-RT and Riverbed Modeler. The false telecontrol commands are first generated using OPAL-RT simulator and saved using the Wireshark tool; Wireshark is an open-source software that is able to monitor, and save communication packets. Tcpreplay command is used to replay the false telecontrol commands using wireshark tool to the riverbed modeler through the "attacker SITL Port" as illustrated in Fig. 5.

As illustrated in Fig. 6, circuit breakers CB10 and CB11 are opened at $t = 1$ s by false telecontrol commands. Therefore, cyberattackers are able to successfully isolate the power plant from the transmission system when station bus in substation A is based on legacy communication networks. It is worth noting that Fig. 6 illustrates the change of the circuit breaker statuses in time as observed by Substation A's operator who is oblivious to communication latency on the wide-area network. Hence, the communications latency is not represented in this figure.

5.2 Case Study II: False Telecontrol Command Injection Attacks Against Power Systems – SDN Communication Networks

In this case study, the station bus in substation A is implemented in Riverbed Modeler based on software-defined networking as illustrated in Fig. 7. At the beginning of the simulation, SDN switches flow table must be initialized. For that purpose, an HTTP request is sent from postman application to the virtual machine that runs the Opendaylight controller. The HTTP request carries a XML body that describes the name of the switch to be updated, the source and destination MAC address of the packets of interest, the type (GOOSE) and the port toward which the packets is to be forwarded. This information is received at the controller level and then converted to an Openflow packets that will be sent from the VM to Riverbed Modeler through SDN controller SITL port as illustrated in Fig. 7. Upon packet receival the targeted switch will update its flow table. It is worth noting that the proactive approach is considered in this paper and SDN controller is disconnected from Riverbed Modeler before co-simulating the test system.

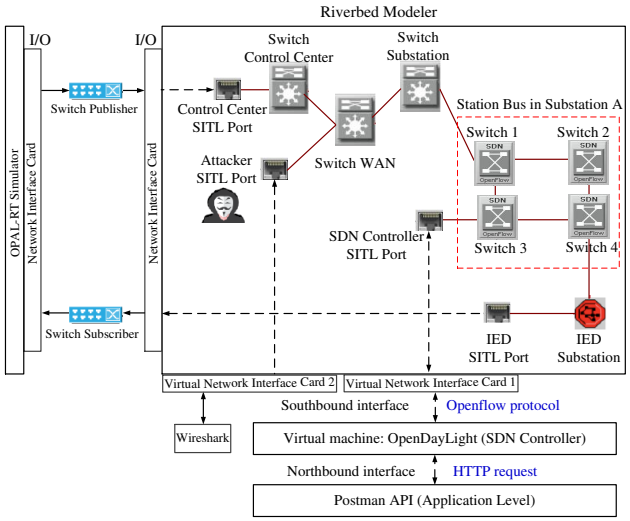

Figure 7: Communication network between the control center and IED in the substation A - station bus based on software-defined networking.

The IEEE PSRC test system is co-simulated again similar to Case Study I and false telecontrol commands are injected into the wide area network through a SITL port, *i.e.*, attacker SITL port, using the Wireshark tool. In this case study, false telecontrol commands are dropped by SDN switches since the source MAC address of false telecontrol commands did not match the forwarding rules in SDN switches and therefore, the cyberattack was unsuccessful. As

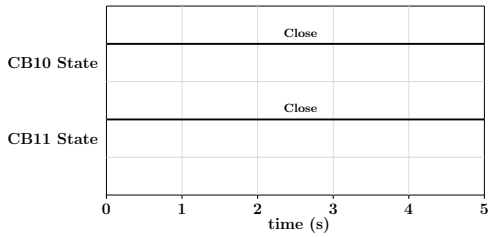

Figure 8: Status of circuit breakers CB10 and CB11 in Case Study II.

illustrated in Fig. 8, circuit breakers CB10 and CB11 remain closed in this case study.

6 CONCLUSION

This paper examined cyberattacks against power systems using false telecontrol commands. Software-defined networking is then proposed as a solution for enhancing cyber-resilience of power systems against false telecontrol commands. It is shown that software-defined networking with whitelisting and deny-by-default features can significantly enhance the cyber-resilience of power systems against cyberattacks using false telecontrol commands. The findings of the paper are tested and validated by a state of the art co-simulator based on OPAL-RT simulator Riverbed Modeler and opendaylight SDN controller. In our future work we will investigate the scalability of SDN technologies in the context of WANs as well as the SDN cybersecurity challenges for power grid applications.

REFERENCES

[1] K. P. Brand, V. Lohmann, and W. Wimmer, Substation Automation Handbook, Utility Automation Consulting Lohmann, 2003.
[2] National Academies of Sciences, Engineering, and Medicine, Communications, Cyber Resilience, and the Future of the U.S. Electric Power System: Proceedings of a Workshop, The National Academies Press, Washington DC, 2020.
[3] National Academics of Sciences, Engineering, Medicine, Enhancing the Resilience of the Nation's Electricity System, National Academics Press, 2017.
[4] T. Smith, Hacker Jailed for Revenge Sewage Attacks, Oct. 2001 [Online]. Available: https://www.theregister.co.uk/2001/10/31/hacker_jailed_for_revenge_sewage/.
[5] G. Liang, S. R. Weller, J. Zhao, F. Luo, and Z. Y. Dong, "The 2015 ukraine blackout: Implications for false data injection attacks," *IEEE Trans. Power Syst.*, vol. 32, no. 4, pp. 3317–3318, 2016.
[6] Industrial Control Systems Cyber Emergency Response Team (ICSCERT) Cyber-Attack Against Ukrainian Critical Infrastructure, 2016 (accessed August 29, 2019).
[7] D. Pliatsios, P. Sarigiannidis, T. Lagkas and A. G. Sarigiannidis, "A survey on SCADA systems: secure protocols, incidents, threats and tactics," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1942–1976, third quarter 2020.
[8] C. Ten, C. Liu and G. Manimaran, "Vulnerability assessment of cybersecurity for SCADA systems", *IEEE Trans. Power Systems*, vol. 23, no. 4, pp. 1836–1846, Nov. 2008
[9] C. Ten, C. Liu and M. Govindarasu, "Vulnerability assessment of cybersecurity for SCADA systems using attack trees," *In Proc. 2007 IEEE Power Eng. Society Gen. Meet.*, Tampa, FL, 2007, pp. 1–8.
[10] Y. Zhang, L. Wang, Y. Xiang and C. Ten, "Power system reliability evaluation with SCADA cybersecurity considerations," *IEEE Trans. Smart Grid*, vol. 6, no. 4, pp. 1707–1721, July 2015.
[11] H. Lin, A. Slagell, C. Di Martino, Z. Kalbarczyk, and R. K. Iyer, "Adapting Bro into SCADA: Building a specification-based intrusion detection system for the DNP3 protocol," *In Proc. 8th Annu. Cyber Sec. Inf. Intell. Res. Workshop (CSIIRW)*, Oak Ridge, TN, USA, 2013, pp. 17-âĂŞ20.
[12] H. Lin, A. Slagell, Z. Kalbarczyk, P. W. Sauer, and R. K. Iyer, "Semantic security analysis of SCADA networks to detect malicious control commands in power grids," *In Proc. Smart Energy Grid Sec. Workshop (SEGS)*, Berlin, Germany, 2013, pp. 29-âĂŞ34.
[13] H. Lin, A. Slagell, Z. T. Kalbarczyk, P. W. Sauer and R. K. Iyer, "Runtime semantic security analysis to detect and mitigate control-related attacks in power grids," *In Proc. IEEE Trans. Smart Grid*, vol. 9, no. 1, pp. 163–178, Jan. 2018.
[14] J. Hong, R. F. Nuqui, A. Kondabathini, D. Ishchenko and A. Martin, "Cyber Attack Resilient Distance Protection and Circuit Breaker Control for Digital Substations," *IEEE Trans. Indust. Informat.*, vol. 15, no. 7, pp. 4332–4341, July 2019.
[15] J. Hong, and C. C. Liu, "Intelligent electronic devices with collaborative intrusion detection systems", *IEEE Trans Smart Grid*, vol. 10, no. 1, pp. 271âĂŞ-281, Jan. 2019.
[16] National Research Council, "Disaster Resilience: A National Imperative", National Academics Press, Washington, D. C., 2012.
[17] D.Jin, Z. Li, C. Hannon, C. Chen, J. Wang, M. hahidehpour and C.W. Lee, "Toward a Cyber Resilient and Secure Microgrid Using Software-Defined Networking", *IEEE Transactions on Smart Grid*, vol. 8, no. 5, pp. 2494-2504, Sept. 2017
[18] D. Ibdah, M. Kanani, N. Lachtar, N. Allan and B. Al-Duwairi, "On the security of SDN-enabled smartgrid systems", *2017 International Conference on Electrical and Computing Technologies and Applications (ICECTA)*, pp. 1–5, Nov. 2017.
[19] M. H. Rehmani, A. Davy, B. Jennings and C. Assi, "Software defined networks-based smart grid communication: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2637âĂŞ- 2670, 2019.
[20] D. Kreutz, F. M. V. Ramos, P. E. Verssimo, C. E. Rothenberg, and S. Azodolmolky, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14-âĂŞ76, 2014.
[21] R. Bobba, D. R. Borries, R. Hilburn, J. Sanders, M. Hadley, and R. Smith, "Software-defined networking addresses control system requirements," April 2014. [Online]. Available: https://selinc.com.
[22] X. Dong, H. Lin and R. Tan, R. K. Iyer, and Z. Kalbarczyk "Software-defined networking for smart grid resilience: opportunities and challenges", *In Proc. 1st ACM Workshop Cyber-Phys. Syst. Secur. (CPSS)*, pp. 61–68, Apr. 2015.
[23] S. M. Mohseni-Bonab, A. Hajebrahimi, I. Kamwa and A. Moeini, "Transmission and distribution co-simulation: a review and propositions," *IET Generation, Transmission & Distribution*, vol. 14, no. 21, pp. 4631–4642, Oct. 2020.
[24] H. Gras, J. Mahseredjian, E. Rutovic, U. Karaagac, A. Haddadi, O. Saad, I. Kocar, and A. El-Akoum, "A new hierarchical approach for modeling protection systems in EMT–type software," *Intern. Conf. Power Syst. Transients*, Seoul, Republic of Korea, June 2017.
[25] IEC TR 61850âĂŞ-90âĂŞ-12:2015, Communication networks and systems for power utility automation âĂŞ- Part 90–12: Wide area network engineering guidelines, Technical Report, July 2015. Available: http://webstore.iec.ch/.