Software Defined Networking opportunities for intelligent security enhancement of Industrial Control Systems

Markel Sainz, Mikel Iturbe, Iñaki Garitano, and Urko Zurutuza

Electronics and Computing Department
Mondragon University, Goiru 2, 20500 Arrasate-Mondragón, Spain,
{msainzo,miturbe,igaritano,uzurutuza}@mondragon.edu

Abstract. In the last years, cyber security of Industrial Control Systems (ICSs) has become an important issue due to the discovery of sophisticated malware that by attacking Critical Infrastructures, could cause catastrophic safety results. Researches have been developing countermeasures to enhance cyber security for pre-Internet era systems, which are extremely vulnerable to threats. This paper presents the potential opportunities that Software Defined Networking (SDN) provides for the security enhancement of Industrial Control Networks. SDN permits a high level of configuration of a network by the separation of control and data planes. In this work, we describe the affinities between SDN and ICSs and we discuss about implementation strategies.

Keywords: Software Defined Networking, Industrial Control Systems, Security, Anomaly Detection

1 Introduction

Since the interconnection of industrial control systems (ICSs) to the Internet, Cyber Physical Systems (CPSs) security has become an important issue. The fact that most of ICSs are composed of legacy equipment, designed in the pre-Internet era, expose them to numerous cyber threats [1]. Not only traditional IT cyber attacks such as DoS or Eavesdropping have been used against ICSs. In 2010, Stuxnet [2] worm demonstrated how sophisticated an attack could be by uploading malicious code to Programmable Logic Controllers (PLCs) and hiding the modifications. After Stuxnet, other ICS oriented malware has been discovered in different facilities. Examples of known worms are NightDragon [3], Duqu [4], Flame [5], Gauss [6] and DragonFly [7]. A set of causes that make ICSs vulnerable are described by Graham *et al.* [8] such as the long hardware replacement periods and their limited computing power, the delay or non-existence of software or firmware updates and patches, the use of insecure communication protocols and the long lasting conviction that security can be enhanced through obscurity.

Software Defined Networking (SDN) has demonstrated benefits in Traffic Engineering (TE) in traditional IT networks [9]. However, SDN has been barely used with cyber security purposes in ICSs. The repetitive network behaviour that characterizes ICSs makes them a good candidate to test the possibility of using SDN in order to develop effective intelligence able to restrict network traffic and to detect anomalies in a reliable manner, concluding in the enhancement of cyber security in ICS networks. This paper approaches the possibility of using SDN with the mentioned purpose. Sections 2 and 3 introduce ICSs, describing their architecture and evolution. Section 4 defines SDN and relates it to ICSs. Section 5 discusses some possibilities among the combination of SDN and ICSs for security purposes. Finally, Section 6 provides some conclusions.

2 Overview of ICS

Industrial Control Systems are a group of ad-hoc elements used for the management of industrial automation systems, with the aim of controlling and monitoring them remotely. Industrial systems, with emphasis in Critical Infrastructures (CIs), are nowadays imperative for life-sustainability and technological and social development. Moteff *et al.* [10] define CIs as

"Infrastructures so vital that their incapacitation or destruction would have a debilitating impact on defense or economic security".

Similarly, Ten *et al.* [11] define them as

"Complex physical and cyber based systems that form the lifeline of a modern society , and their reliable and secure operation is of paramount importance to national security and economic vitality".

Examples of critical infrastructures include power generation stations, water supply plants and manufacturing industries. Due to the inter-dependability among CIs [12], a malfunction in a particular plant can compromise other infrastructures, becoming a potential risk which could cause catastrophic consequences.

2.1 Network architecture

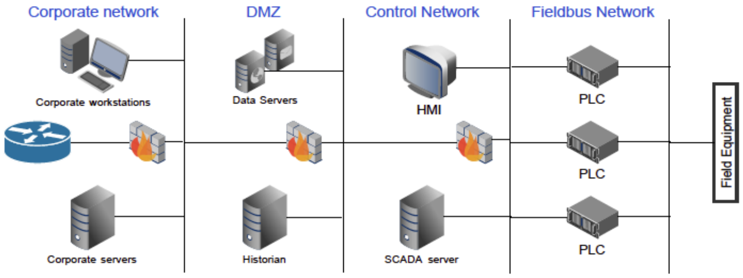

Fig. 1. Example of Industrial Network Architecture.

The main objective of an ICS network is to manage and monitor physical assets. ICS components can be classified into the following three groups: Field Devices (Sensors and Actuators), Field Controllers (PLCs, RTUs and IEDs) and Control/Supervisory devices (MTUs, HMIs and Historians). For a more detailed description of the assets, refer to [13] and [14].

Figure 1 shows a typical ICS-IT network architecture, designed having as a reference the work presented by Krotofil *et al.* [1] and Galloway *et al.* [13]. The network topology from Figure 1 is divided in layers, each of them representing a different section of the network where different kind of ICS components can be found. The outer (left) layer is connected to the Internet through the corporate IT network. The inner layer, represents an ICS network composed by physical elements and logical controllers.

The corporate network layer represents a traditional IT network, where the regular corporation assets are located, such as servers and computers. In the next network layer, the Demilitarized Zone (DMZ) is placed. The DMZ layer acts as an intermediate layer between the control network and the corporate network, in order to prevent direct access from corporation assets to the control network. The devices placed in the DMZ are generally data servers which must be accessed both from control network and corporate network. For example, corporate network may need to use historian data for statistical analysis at the same time that control network registers data on it.

The Fieldbus network layer is composed by the field controllers, which, in essence, manage sensors and actuators. In the control network, both supervisory devices and field controllers are interconnected, and the last ones are directly connected to the Fieldbus Network. Although Figure 1 shows a simple scenario, control networks can be much more complex, for instance, with the addition of slave RTUs.

Finally, directly connected to the Fieldbus network layer, field equipment can be found. These devices, send and receive data from control devices in order to inform about the industrial environment situation and actuate adequately to it.

It is necessary to add that this figure represents a possibility among a wide variety of implementations. Unlike IT networks, which are generally composed by Ethernet and WIFI connections, ICSs nature tends to be more heterogeneous, especially when involving field assets. In the lower layers of the architecture several types of connections can be found, such as Ethernet, Serial and some other field buses.

3 Evolution of cyber security demands in industrial systems

In the past decades, industry and automation has been spread all around the world in a massive way. Nowadays, industry can be considered the engine of the society for two reasons: the supply of needed amount of productivity due to increasing demand and the creation of jobs or employment opportunities.

When industrialization started, automation systems worked in an isolated way, due to the lack of necessity of intercommunication. Due to the increasing development of IT technologies, and the need of communicating industrial data through long distances, automation systems started to open to the Internet [13]. Thus, this event enabled the possibility to control industrial systems remotely and to interconnect remote sites, introducing concepts such as Distributed Control System (DCS) and Supervisory Control and Data Acquisition (SCADA), improving efficiency and easing data collection in order to be processed. On the other hand, connecting a device to the Internet means therefore, making it vulnerable to security threats [1].

Consequently, after realizing the potential threat, the scientific community has been working on different approaches to enhance cyber security in critical infrastructures. Due to their focus in availability, it is difficult to replace old equipment for a newer or modern one, the development of technologies able to cope with legacy devices has been necessary.

The services provided by IT and industrial networks have their security basis in a concept called CIA triad. This acronym refers to Confidentiality, Integrity and Availability. These requirement must be satisfied by an ICS to consider it secure. According to Cheminod *et al.* [15] the mentioned security requirements can be defined as follows:

- Confidentiality: It ensures the information available in a system is not revealed to any person, entity or process that has not the necessary authorization.
- Integrity: It refers to the ability of preventing unauthorized and undetected modification of the information.
- Availability: Guarantees the information is accessible for authorized users by preventing possible access deny attacks from unauthorized users.

Industrial equipment is composed mainly by legacy equipment which has longer life-cycle than IT due to it's reliability requirements [16]. This leads to the existence of multiple different technologies composing ICSs, such as Operating Systems, Network protocols and hardware. This heterogeneity makes difficult the enhancement of cyber security.

4 SDN benefits in securing ICS

Software Defined Networking is a relatively new networking paradigm that separates control and data plane, in order to ease the management and maintenance of IT networks [17]. Thus, network behaviour becomes programmable by a centralized controller, while network elements forward traffic according to established flow-tables or rule sets. Traditional IP networks are designed in a way where logic is distributed among all the network elements, forcing them to forward traffic according to packet's destination address and acting as independent devices which have strongly limited visibility of the rest of the network [18]. In the case of SDN, the behaviour of each network device is defined by software in a network controller, which then transmits to the data plane devices the corresponding flow-tables. If a switch receives a packet which can't forward due to a rule lack, it can communicate with the controller to resolve the issue.

4.1 SDN research

Software Defined Networking has been widely adopted with Traffic Engineering (TE) purposes in the last years as stated by Mendiola *et al.* [9]. It has demonstrated interesting capabilities in performance optimization of wide IT networks, that is why renowned entities such as Google have implemented it in their WANs [19]. Mousa *et al.* [18] refer to some SDN applications in IT network security such as NICE [20], FlowGuard [21] and sFlow [19]. Regarding ICSs, SDN has been barely used to enhance cyber security. Molina *et al.* [22] describe an implementation of SDN in ICSs based in IEC 61850 for TE with interesting security aspects. The authors propose security improvement in three different ways. Traffic isolation it's been traditionally done by employing VLAN (IEEE 802.1Q), limiting the broadcasting range to a single network. Molina *et al.* describe the use of a Virtual Network Filter Module, able of creating logical networks based on MAC addresses, avoiding the need to use VLANs. For anomaly detection, they encourage the use of sFlow, establishing desired network behaviour thresholds and communicating the controller if they are exceeded. The platform permits the introduction of flows based on different parameters (MAC/IP addresses, Ethertype, VLAN, TCP/UDP ports...) and monitoring them. Thresholding has demonstrated being useful against DoS and DDoS attacks, due to the possibility of altering flow tables in real time as a countermeasure. Lastly, the authors propose the use of a Firewall module to limit ingress traffic by MAC source address, port and switch. This way, resilience is gained against MAC spoofing attacks. Dong *et al.* [23] describe the opportunities provided by SDN for smart grid resilience. They mention the possibility of dynamically configuring policies to filter unwanted or potentially malicious traffic due to the compromise of switches, grid devices, RTUs, SCADA slaves, etc. Moreover, switches can be configured in execution time to enable dynamic monitoring of suspiciously excessive traffic towards a concrete destination. They also encourage the use of Virtual Network Layering and they describe the capacity of hot-swapping between private and public networks. This last aspect may be crucial when under attack, due to the possibility of redirecting critical traffic through the Internet when local network is highly compromised.

4.2 Exploitable affinities

In the last decades, Intrusion Detection Systems (IDSs), in conjunction with recommended security practices, have been developed and used to secure critical infrastructures. IDSs can be categorized in two main groups: signature based and anomaly based [24]. The first group is highly effective in the detection of previously registered signatures of known malware, while being useless for zero-day attacks. On the other hand, anomaly based IDS monitor network packets to capture uncommon behaviours. These last IDSs are able to detect zero-day attacks, but false positive rates can be high. Network IDSs (NIDS) are placed somewhere in a LAN/WAN and collect traffic to analyse, so they are quite limited devices due to the low visibility of the rest of the network [17].

Anomaly based IDS base their accuracy in the high periodicity of ICS networks [25]. Unlike IT networks, where traffic patterns are very variable due to the dependence of user behaviour, communication between devices in ICS networks occurs in a pre-established way in most cases. The communication among devices in an ICS network occurs mainly in the following way:

1. SCADA server sends a request to a PLC where the value of a variable or group of variables is solicited.
2. The PLC receives the request and processes it, collecting the necessary data from sensors and sending a response to the server with the requested data.
3. SCADA server receives the response and stores the data. A Human Machine Interface (HMI) can pull the data from the server to inform the operators of the system's state. In case there is a historian, a similar transmission will be done to register collected data.

The communication between low level devices such as PLCs is generally not necessary and the pattern will only change if a specific order is introduced by an operator, such as changing the state of an actuator. The automatic communications in an ICS network will occur in pre-established time windows, that is the reason why the high periodicity and determinism of these kind of systems can be considered key values in anomaly detection mechanisms.

In a similar way to the functioning of IDSs, SDN switches register every packet they forward, being able to send traffic statistics to their controllers. Moreover, SDN switches can be configured in order to deny any traffic not included in their flow tables, which grants a high level isolation between devices. Having this in mind, it is possible to configure a switch in order to route packets not only by source and destination addresses, but by ingress port or header and payload content. Taking into account that network traffic in an ICS network is known and periodic, flow-entries can be established statically before execution. If a switch receives a packet that cannot forward due to lack of rules, the device will ask the controller for a new flow-entry in order to forward the packet correctly. Flow-entries can be marked as static so that if the controller crashes, switches can continue operating. Although SDN may provide security enhancements, SDN-capable devices are likely to suffer from vulnerabilities. Several countermeasures are proposed by Kreutz *et al.* [26] and Dabbagh *et al.* [17] such as controller replication in conjunction with platform diversity and voting mechanisms in case a controller gets compromised, and message-length and inter-packet arrival time definition for encrypted or tunnelled packet forwarding.

4.3 Protocols and Experimentation tools

SDN paradigm is implemented by numerous communication protocols nowadays. These protocols can be categorized in three groups according to [9]:

- D-CPI protocols: They are used to communicate data and control planes. They contain information about data plane resources and possible operations. In this layer protocols such as OpenFlow [27], ForCES [28], I2RS [29] and BGP-LS/PCEP [30] can be found. OpenFlow has gone through six revisions since its launch and has been widely used in IT networks due to the high rate of deployment of networking vendors [17].
- A-CPI protocols: Their objective is to provide a communication layer between the controller and the applications running over it. In this group, ALTO [31] must be considered. This protocol provides a suitable API that contains information about the state of the network in order to improve applications and network performance.
- MI protocols: This last group is in charge of network configuration through all planes, focusing mainly in the management of network elements. Protocols included in this layer are Open vSwitch Database Management (OVSDB) [32], OpenFlow Configuration (OF-CONFIG) [33] and NETCONF [34].

Taking into account the nature of automation systems, there is little to no possibility of testing new technologies on real operating environments. To solve this issue, Antonioli *et al.* propose MiniCPS [35], a set of Python tools to simulate Cyber Physical Systems such as ICSs. MiniCPS uses Mininet [36] to emulate network elements, and ICS components such as PLC are defined by Python scripts. This tool can be used to test different SDN protocols, developing the needed functional intelligence on top of them.

5 Discussion and future research lines

After having noted the SDN potential, it can be deducted that the filtering capabilities in conjunction with the high level of monitoring provided can be decisive in attack detection and mitigation. No research work has been done in the use of machine learning along with SDN for security in ICS. We propose the development of required intelligent modules on top of the controller to provide the security mechanisms described below. Firstly, the traffic filtering capabilities have to be used, limiting packets by different header values, payload content or message length, ingress port and source/destination address and arrival times. As the traffic behaviour in ICSs is known, the rules can be defined before analysing the network pattern. Flow tables that define the mentioned restrictions should be dynamically configurable in order to create time-window restricted flows to permit the interaction with authorized operators. Anyway, it is necessary to add the possibility of marking static flow-entries in case of controller crash. For the detection of attacks, traffic statistic recollection capabilities can be used. An application in the controller can be created which will initially construct a normal behaviour pattern observing the entire network statistic sent from switches under normal circumstances, in which will also be included a behaviour model obtained in a pre-established operator interaction time-window. Once the pattern is created, previously mentioned parameters alteration could be detected. Giotis *et al.* [19] propose the utilisation of sFlow due to the limited capacity that forwarding devices can have for storing rules and packet counts. In the case of ICSs, packet count of permitted flows will be high, while rest of flow counts low, so excessive potentially malicious traffic could be detected rapidly and overload avoided by resetting affected packet counters and deleting unused flow-entries after pre-established periods of time. In case the device which receives the malicious packet(s) is not able of denying the attack, dynamical configuration alteration should be supported in order to drop packets or change routing. Dong *et al.* [23] mention the possibility of swapping to public networks when a big part of the network is compromised. We propose to add notification capabilities to the controllers so that when an anomaly is detected in a switch, the controller will be communicated and this, at the same time, will send an alert to the SCADA server or an HMI, in order to notify of the issue and permit operators actuate consequently. This notification can be sent via mail or phone, but having in mind the additional vulnerabilities and threat vectors this added functionalities can bring in, specific countermeasures and isolation mechanism have to be designed in order to avoid any unwanted interaction with the industrial network. According to Molina *et al.* [22], traffic isolation modules are supported by SDN protocols. The logical isolation of actuators could prevent important damages in case of network break, by denying broadcasting and permitting communication from allowed devices only.

6 Conclusions

With the increasing propagation of SDN protocols use, research in the suitability of different purposes is being carried out. This work has analysed the potential affinity between ICSs nature and SDN technology for security purposes. Until the moment, little research has been done in this area, so the suitability on production environments has not been tested yet. The utilisation of SDN capable networking equipment can help enhance security with low performance impact and low investment. On the other hand, due to the characteristic heterogeneity present in ICS networks, further research has to be done to test the viability of the technology. Nowadays, IDSs are used to detect attacks, which require dedicated equipment and processing capabilities. SDN provides a similar capability in conjunction with traffic filtering, probably using the same forwarding devices present in many networks. Additionally, SDN provides a layer of prevention due to the high network configuration and visibility of the entire network that permits. The experimentation in this area with MiniCPS will possibly lead to concluding results and new security mechanisms on top of SDN.

Acknowledgements This work has been developed by the intelligent systems for industrial systems group supported by the Department of Education, Language policy and Culture of the Basque Government. It has been partially funded by the European Union's Horizon 2020 research and innovation programme's project MC-SUITE under grant agreement No 680478.

References

1. M. Krotofil and D. Gollmann, “Industrial control systems security: What is happening?” in *2013 11th IEEE International Conference on Industrial Informatics (INDIN).* IEEE, 2013, pp. 670–675.
2. S. Karnouskos, “Stuxnet worm impact on industrial cyber-physical system security,” in *IECON 2011-37th Annual Conference on IEEE Industrial Electronics Society.* IEEE, 2011, pp. 4490–4494.
3. G. E. Cyberattacks, “Night dragon,” *McAfee Foundstone Professional Services and McAfee Labs*, 2011.
4. B. Bencsáth, G. Pék, L. Buttyán, and M. Félegyházi, “Duqu: Analysis, detection, and lessons learned,” in *ACM European Workshop on System Security (EuroSec)*, vol. 2012, 2012.
5. K. Munro, “Deconstructing flame: the limitations of traditional defences,” *Computer Fraud & Security*, vol. 2012, no. 10, pp. 8–11, 2012.
6. B. Bencsáth, G. Pék, L. Buttyán, and M. Felegyhazi, “The cousins of stuxnet: Duqu, flame, and gauss,” *Future Internet*, vol. 4, no. 4, pp. 971–1003, 2012.
7. N. Nelson, “The impact of dragonfly malware on industrial control systems,” *SANS Institute*, 2016.
8. J. Graham, J. Hieb, and J. Naber, “Improving cybersecurity for industrial control systems,” in *Industrial Electronics (ISIE), 2016 IEEE 25th International Symposium on.* IEEE, 2016, pp. 618–623.
9. A. Mendiola, J. Astorga, E. Jacob, and M. Higuero, “A survey on the contributions of software-defined networking to traffic engineering,” *IEEE Communications Surveys & Tutorials*, 2016.
10. J. Moteff, C. Copeland, and J. Fischer, “Critical infrastructures: What makes an infrastructure critical?” DTIC Document, 2003.
11. C.-W. Ten, G. Manimaran, and C.-C. Liu, “Cybersecurity for critical infrastructures: Attack and defense modeling,” *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 40, no. 4, pp. 853–865, 2010.
12. K. Stouffer, J. Falco, and K. Scarfone, “Guide to industrial control systems (ics) security,” *NIST special publication*, vol. 800, no. 82, pp. 16–16, 2011.
13. B. Galloway and G. P. Hancke, “Introduction to industrial control networks,” *IEEE Communications surveys & tutorials*, vol. 15, no. 2, pp. 860–880, 2013.
14. P. Eden, A. Blyth, P. Burnap, Y. Cherdantseva, K. Jones, H. Soulsby, and K. Stoddart, “A cyber forensic taxonomy for scada systems in critical infrastructure,” in *International Conference on Critical Information Infrastructures Security.* Springer, 2015, pp. 27–39.
15. M. Cheminod, L. Durante, and A. Valenzano, “Review of security issues in industrial networks,” *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 277–293, 2013.
16. K. Kobara, “Cyber physical security for industrial control systems and iot,” *IEICE TRANSACTIONS on Information and Systems*, vol. 99, no. 4, pp. 787–795, 2016.
17. M. Dabbagh, B. Hamdaoui, M. Guizani, and A. Rayes, “Software-defined networking security: pros and cons,” *IEEE Communications Magazine*, vol. 53, no. 6, pp. 73–79, 2015.
18. M. Mousa, A. M. Bahaa-Eldin, and M. Sobh, “Software defined networking concepts and challenges,” in *Computer Engineering & Systems (ICCES), 2016 11th International Conference on.* IEEE, 2016, pp. 79–90.

19. K. Giotis, C. Argyropoulos, G. Androulidakis, D. Kalogeras, and V. Maglaris, “Combining openflow and sflow for an effective and scalable anomaly detection and mitigation mechanism on sdn environments,” *Computer Networks*, vol. 62, pp. 122–136, 2014.
20. C.-J. Chung, P. Khatkar, T. Xing, J. Lee, and D. Huang, “Nice: Network intrusion detection and countermeasure selection in virtual network systems,” *IEEE transactions on dependable and secure computing*, vol. 10, no. 4, pp. 198–211, 2013.
21. H. Hu, W. Han, G.-J. Ahn, and Z. Zhao, “Flowguard: building robust firewalls for software-defined networks,” in *Proceedings of the third workshop on Hot topics in software defined networking.* ACM, 2014, pp. 97–102.
22. E. Molina, E. Jacob, J. Matias, N. Moreira, and A. Astarloa, “Using software defined networking to manage and control iec 61850-based systems,” *Computers & Electrical Engineering*, vol. 43, pp. 142–154, 2015.
23. X. Dong, H. Lin, R. Tan, R. K. Iyer, and Z. Kalbarczyk, “Software-defined networking for smart grid resilience: Opportunities and challenges,” in *Proceedings of the 1st ACM Workshop on Cyber-Physical System Security.* ACM, 2015, pp. 61–68.
24. V. Jyothsna, V. R. Prasad, and K. M. Prasad, “A review of anomaly based intrusion detection systems,” *International Journal of Computer Applications*, vol. 28, no. 7, pp. 26–35, 2011.
25. A. Kleinman and A. Wool, “Accurate modeling of the siemens s7 scada protocol for intrusion detection and digital forensics,” *The Journal of Digital Forensics, Security and Law: JDFSL*, vol. 9, no. 2, p. 37, 2014.
26. D. Kreutz, F. Ramos, and P. Verissimo, “Towards secure and dependable software-defined networks,” in *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking.* ACM, 2013, pp. 55–60.
27. N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, “Openflow: enabling innovation in campus networks,” *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 69–74, 2008.
28. A. Doria, J. H. Salim, R. Haas, H. Khosravi, W. Wang, L. Dong, R. Gopal, and J. Halpern, “Forwarding and control element separation (forces) protocol specification,” Tech. Rep., 2010.
29. S. Hares and R. White, “Software-defined networks and the interface to the routing system (i2rs),” *IEEE Internet Computing*, vol. 17, no. 4, pp. 84–88, 2013.
30. H. Gredler, J. Medved, S. Previdi, A. Farrel, and S. Ray, “North-bound distribution of link-state and traffic engineering (te) information using bgp,” Tech. Rep., 2016.
31. R. Alimi, Y. Yang, and R. Penno, “Application-layer traffic optimization (alto) protocol,” 2014.
32. B. Pfaff and B. Davie, “The open vswitch database management protocol,” 2013.
33. R. Narisetty, L. Dane, A. Malishevskiy, D. Gurkan, S. Bailey, S. Narayan, and S. Mysore, “Openflow configuration protocol: implementation for the of management plane,” in *Research and Educational Experiment Workshop (GREE), 2013 Second GENI.* IEEE, 2013, pp. 66–67.
34. R. Enns, “Netconf configuration protocol,” 2006.
35. D. Antonioli and N. O. Tippenhauer, “Minicps: A toolkit for security research on cps networks,” in *Proceedings of the First ACM Workshop on Cyber-Physical Systems-Security and/or PrivaCy.* ACM, 2015, pp. 91–100.
36. M. Team, “Mininet: An instant virtual network on your laptop (or other pc),” 2012.