Intrusion and Anomaly Detection in Industrial Automation and Control Systems

Luis Rosa, Tiago Cruz, Paulo Simões and Edmundo Monteiro
University of Coimbra, CISUC, DEI
Email: {lmrosa, tjcruz, psimoes, edmundo}@dei.uc.pt

***Abstract*—In the domain of Industrial Automation and Control Systems (IACS), security was traditionally downplayed to a certain extent, as it was originally deemed an exclusive concern of Information and Communications Technology (ICT) systems. The myth of the air-gap, as well as other preconceived notions about implicit IACS security, constituted dangerous fallacies that were debunked once successful attacks become known. Ultimately, the industry started shifting away from this dangerous mindset, discussing how to properly secure those systems. In many ways, IACS security should not be treated differently from modern ICT security. For sure, IACS have distinct characteristics, assets, protocols and even priorities that should be considered – but security should never be an optional concern.**

In this publication, we present the main results of a PhD dissertation that proposes a holistic and data-driven framework capable of leveraging distinct techniques to increase situational awareness and provide continuous and near real-time monitoring of IACS. For such purposes, it proposes an evolution of the Security Information and Event Management (SIEM) concept, geared towards providing a unified security data monitoring solution by leveraging recent advances in the field of real-time Big Data analytics. In the same way, the most recent machine-learning-based anomaly-detection techniques (which are becoming increasingly prominent in the cybersecurity field) are also analyzed and studied to understand their benefits for developing and advancing IACS cyber-intrusion detection processes.

***Index Terms*—Industrial Automation and Control Systems; Cybersecurity; Intrusion Detection; Real-Time Big Data Analytics; SCADA Networks.**

I. Introduction

Industrial Automation and Control Systems (IACS) [1] [2] differ from Information Technology (IT) systems. A successful cyber-attack against mission-critical IACS can lead to massive financial losses, physical equipment damage or even human safety hazards. The cybersecurity of IACS, sometimes overlooked in the past, is now a paramount matter. But this is no easy task, for reasons such as the widespread usage of legacy technologies and protocols or the number of legacy systems still in use without proper security support, often operating beyond the equipment's End-Of-Life support status.

IACS traditionally relied on air-gapped Supervisory Control And Data Acquisition (SCADA) systems using domain-specific protocols and technologies. The increased interconnection paths, the Operational Technology (OT)/IT convergence and the gradual adoption of Ethernet- and TCP/IP-based networks in IACS faded the perimeter lines between what was assumed to be secure and the outside world. This is aggravated by the fact that most of the SCADA communication protocols still lack proper security enforcing mechanisms, despite the gradual introduction of some security support.

Contrary to what could be expected, the support for classical tools such as rule-based Network Intrusion Detection System (NIDS) for SCADA is limited. Even the most popular NIDS are often limited to *ad-hoc* rules for SCADA traffic, lacking the appropriate stateful/stateless decoding capabilities and richer signature sets for the different protocols. Anomaly detection based on Machine-Learning (ML) techniques, increasingly prominent in other domains, is also expected to bring numerous benefits to IACS, including efficient classification of large amounts of heterogeneous data to spot anomalies. Nevertheless, they are still presented in the literature as theoretical and isolated works focused mostly on proposing and evaluating specific algorithms.

This gap creates the opportunity to introduce evolved SIEM systems as a good match for monitoring and integrating a wide range of additional security mechanisms. More than a single NIDS or a single ML-based anomaly detection algorithm, SIEM systems are expected to bring a global, aggregated and valuable insight into the security state of the infrastructure. Nevertheless, the applicability of SIEMs in SCADA environments is still in its early stages. There are still several open challenges, such as the data formats, integration/interoperability between components or overall platform orchestration, which need to be addressed to achieve practical and complete solutions.

A. Research Question and Contributions

Considering the previously identified problems, the thesis hereby summarized [3] addresses the following research question: *How to improve the security of next-generation IACS through a holistic data-driven framework?* This work led to several contributions, of which we emphasize the following:

- *Security analysis of SCADA protocols in the scope of practical attack scenarios.* Among other specific results, we point out the definition and exploration of practical attack scenarios, often from the attackers' perspective (less used in the literature) and based on testbeds representative of real IACS operated by energy utilities. Besides some initial work focused on the well-known Modbus protocol [4], a detailed security analysis of the PCOM [5] protocol was also conducted. This protocol was chosen as an example of various popular SCADA protocols used by the industry but not extensively studied

from a security point of view. This was validated by leveraging the CockpitCI and ATENA testbeds to create high-fidelity scenarios [6]–[8]. Moreover, this effort also resulted in multiple contributions to popular open-source tools, such as the Snort IDS [9], Wireshark [10]–[12], Scapy [13], NMAP [14] and Metasploit [15], [16], as well as a publicly available dataset [17].

- *Conceptualization and design of a holistic data-driven framework for intrusion and anomaly detection in IACS scenarios.* Leveraging the SIEM and lambda architecture concepts, this conceptualization and design work addresses challenges such as the integration of multiple, dispersed and heterogeneous data sources, platform elasticity and scalability (for being able to ingest large amounts of dispersed data while keeping time-boundaries for data processing within required levels, for streaming and batch processing), and to flexibly accommodate and combine different anomaly detection mechanisms in a neutral fashion [18]–[20].
- *Integration and evaluation of different mechanisms for classifying network traffic.* For demonstration and validation purposes, several network traffic classification mechanisms were integrated into the framework and evaluated, considering various representative scenarios, both for near real-time detection (based on stream processing) and for batch processing [20].

The rest of this paper is organised as follows: Section II describes the performed exploratory analysis of different SCADA protocols, tools and attack scenarios; Section III introduces the proposed holistic Intrusion and Anomaly Detection System (IADS) framework; Section IV addresses the event streaming and data analytics capabilities, two key building blocks of the proposed framework; and Section V concludes the paper.

II. Exploratory Analysis of the Security of SCADA Protocols

In the scope of IACS, network communication protocols are one the most valuable data sources for security purposes. They are used, for instance, to monitor and actively control components from a local/remote site or to continuously poll the values of an autonomous physical process. In that sense, our work started by researching network-based attack scenarios using the Modbus protocol [4]. Rather than exclusively studying its widely known vulnerabilities, Modbus was analysed from a different perspective: how SCADA systems can be effectively exploited from a practical standpoint. These experiments, conducted from the attacker's perspective, allowed us to dig into the technical details of involved protocols and, ultimately, to understand how to design and implement appropriate countermeasures. A grey box penetration testing approach was used, narrowing the experiments to the existing SCADA assets and vulnerabilities rather than blindly exploring other types of attack vectors. An incremental three-stage attack strategy was devised, with the following phases: monitor the process values (to gain knowledge about the nature and characteristics of the controlled process), change them without being noticed in the SCADA Human-Machine Interface (HMI) consoles and, finally, induce service disruption. Such strategy, primarily focused on network communications, also covers a large subset of SCADA specific cyber-attack scenarios – including, amongst others, TCP/IP network scans, Modbus specific scans, different variants of Denial of Service (DoS) attacks, and a SCADA-specific Man-in-the-middle (MitM) attack specifically customized for this process environment.

Next, we devoted efforts towards analysing the security of the PCOM protocol [5]. PCOM is one of the many SCADA protocols used by the industry. It was used in our partner's testbed (which reproduced various smartgrid scenarios) for ancillary functions, and accidentally discovered in network traffic captures during the exploratory work conducted from the attacker's perspective. After discovering PCOM traffic, we realised there was no publicly known research about the security of this SCADA protocol. Furthermore, even Wireshark, for instance, lacked support for PCOM, making it difficult even to understand what was transmitted over the network. Taking all of this into account, we decided to conduct an extensive analysis of PCOM's security, as an example of the work that needs to be performed for a large number of still unaddressed SCADA protocols.

In that scope, we started by developing a Wireshark PCOM/TCP dissector. This dissector can interpret PCOM/TCP headers, as well as the header structure for both modes of PCOM (PCOM/ASCII and PCOM/Binary) and interpret over 25 PCOM command codes. The code of this dissector has been integrated into the upstream Wireshark repository [10] [11] [12]. Developing this built-in dissector for Wireshark provided an interface to visualize PCOM messages' flows, structure and content.

Next, we explored different use cases in the form of cyber-attacks using PCOM, which led to the following open-source contributions: two NMAP scans scripts for collecting PLC data using PCOM and CIP protocols [14], [21]; two Metasploit modules to implement several proof-of-concept attacks and to test PCOM Snort rules [15], [16]; a SCAPY layer to decode, manipulate and craft PCOM network packets [13]; and a Snort Ruleset which adds PCOM signature-based detection capabilities to Snort [9] and can be used to detect and limit unwanted network communications. Moreover, a PCOM traffic dataset was also made publicly available, to support PCOM protocol analysis efforts and tool development processes [17].

Like other contemporary SCADA protocols, it was found that PCOM lacks security features such as confidentiality or integrity. Nevertheless, despite the importance of such issues, it must be clearly stated that PCOM is no worse than its contemporary counterparts.

III. A Holistic Intrusion and Anomaly Detection System for IACS

After the aforementioned exploratory analysis of SCADA protocols, we focused on evolving the way security monitoring is performed in IACS environments.

In the last years, IACS security has been extensively discussed [20], [22]–[26], and security practitioners have been rather vocal about its design flaws. Moreover, there is now extensive literature devoted to the (in-)security of SCADA communication protocols, but it clusters around a small subset of the most used and well-known protocols. The security enforcing mechanisms of many other protocols still lack research and validation, especially for closed or poorly documented protocols.

Most of the reviewed literature on anomaly detection using ML-based techniques focuses on the network or process-related features (cf. [3], Chapter 2). Still, other potentially relevant features, such as diversified log sources and host-based events, are commonly ignored, therefore missing an important opportunity to develop a more comprehensive approach covering a broader spectrum of attacks. Another overlooked aspect of surveyed literature is that as we move towards a Big Data problem, it is also critical that the chosen approaches can fit into scalable, distributed and parallel computation environments.

After analysing multiple approaches, techniques and frameworks for Complex Event Processing and Big-Data, we conceived a domain-specific IADS. The proposed framework was built upon the idea of having an evolved Big Data-like SIEM system capable of detecting in (near) real-time the occurrence of cyber-physical attacks in complex IACS environments. In contrast to other works, this research focused on having a holistic framework capable of supporting different scenarios and techniques, inspired by the Lambda Architecture [18].

This constitutes a departure from our first approach towards the problem, where we devised a Defense-in-Depth (DiD) detection layer based on the idea of monitoring a single infrastructure [6]. Instead, we evolved towards an IADS detection layer [20] designed from scratch to cope with highly distributed, capillary and heterogeneous IACS, such as modern smartgrids and industry 4.0 infrastructures, leveraging the integration of open-source tools to create a Big-data oriented solution. Figure 1 provides an overview of the proposed architecture.

The proposed IADS uses a multi-layer approach, including the two key functional capabilities which were more extensively addressed in our research, namely: Stream Processing, which is provided by the messaging system and the domain processors; and Data Analytics, implemented within the Big Data-like SIEM. Figure 2 illustrates how these two functional capabilities relate.

Stream Processing is ensured by multiple domain processors per domain. Each one, designed as a lightweight event-driven streaming processing task, has its own topology (i.e. a chain of small processing steps). Multiple domain processors and individual tasks can also be composed into complex and per-domain processing schemes. For instance, a chain of small domain processors where the output of each one is processed by the next, thus enabling large scale event processing.

Data Analytics capabilities, implemented within the Big Data-like SIEM, are used to run different anomaly detection

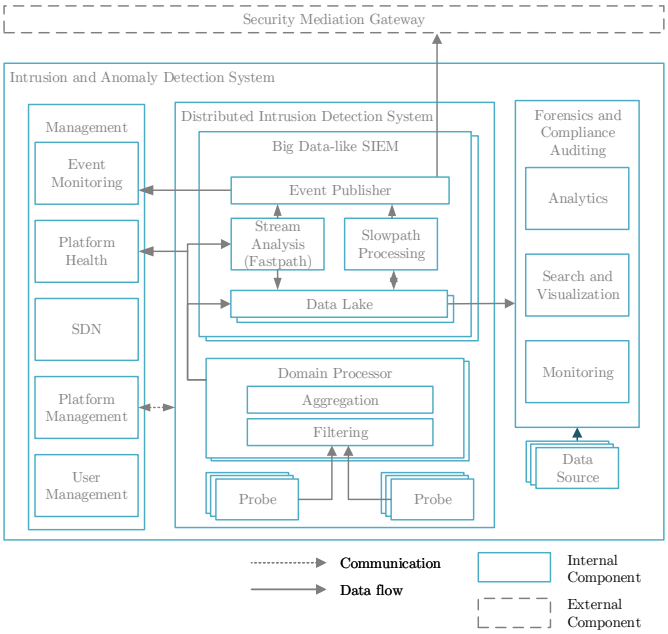

Fig. 1. Intrusion and Anomaly Detection System architecture [19]

mechanisms (i.e., ML-based pipelines). Each pipeline comprises a set of steps, from feature extraction and transformation, up to the actual computation of the probability of the occurrence of a cyber-attack based on previously trained ML-models. Such a generic computation framework is expected to increase overall efficacy in the intrusion detection process and support the handling of different types of cyber-security issues (e.g. computing-intensive algorithms, global deviant patterns, cross-domain incidents, network-based attacks, and physical-based attacks hidden from the network).

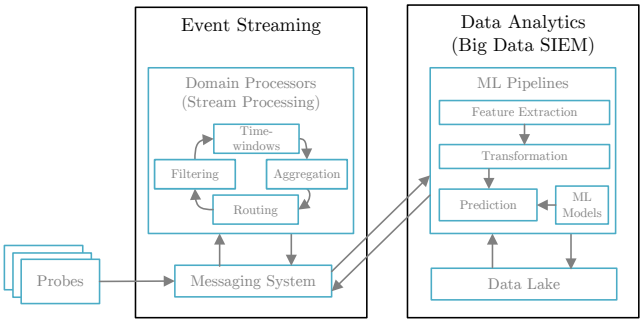

Fig. 2. Stream processing and Data Analytics layers [3].

The proposed framework considers the following principles (cf. [3], Chapter 3): *Fault-Tolerance*; *Effectiveness and Efficiency*; *Scalability*; *Flexibility*; *Security*; *Distributed computation, messaging and storage*; *Node redundancy and strong processing semantics*; *Multiple detection and analytics techniques*; *Stream and Batch Processing*. As opposed to the majority of the surveyed research on IACS security, mainly narrowed to theoretical mechanisms or specific algorithms, the

proposed approach widens the approaches towards effective strategies to materialize and combine different components, mechanisms and tools in the form of a highly flexible and scalable framework.

IV. Event Streaming and Data Analytics

Event streaming, one of the core modules of the proposed framework, fulfils two main purposes: (1) to provide an efficient, distributed and decoupled mechanism for inter-component communication with *exactly-once* processing guarantees; and (2), to provide domain-level processing capabilities. The idea is that different (and heterogeneous) security probes can leverage the event streaming layer to push their outputs and evidence (i.e. events) to a highly flexible messaging system.

Taking the inspiration from Intrusion Detection Message Exchange Format (IDMEF) [27], a custom datamodel was specifically devised for representing generic events (i.e. not only a security event but also events such as telemetry data). Such datamodel is intended to avoid the complexity of other existing formats and fit into the increasingly demanding (Big Data) IACS environments.

Apache Kafka [28] was used to implement the proposed messaging system, since it is a distributed messaging system capable of achieving high message throughput without sacrificing latency – potentially supporting millions of messages per second, as required by larger IACS.

The proposed approach is able to keep up with a substantial number of security probes, as it was designed to scale as needed. Multiple probes may report the same occurrence or attack (e.g. a network scan can be detected, for instance, by a NIDS and a honeypot). Moreover, the same attack may be successively reported by the same probe (e.g. long network-based attacks may get reported multiple times by the same NIDS instance). Some anomaly detection algorithms may not need the output of all probes – probes may generate more data than required (or irrelevant data) for the anomaly detection task. On the other hand, simpler probes may lack contextual information or fail to comply with the event format (e.g. third-party probes hard to customize). In summary, there are several reasons why we need to optimize event flows by grouping them. Many of these scenarios imply the need for some preprocessing.

To address this need, we devised a modular event streaming layer based on a set of domain processors for implementing the first-preprocessing step, decoupled from the remaining components (e.g. a given security probe) and implementing all sorts of messaging and routing patterns (e.g. Content-Based Router, Recipient List, Routing slip, Re-sequencer) [29]. Figure 3 shows an example of how Kafka Streams Domain Specific Language (DSL) [30] is leveraged to implement a feature aggregation task. Such tasks are the foundation of the domain processor concept. In the presented example, the domain processor extracts additional aggregated features grouped by time windows on top of a stream of individual network packet features.

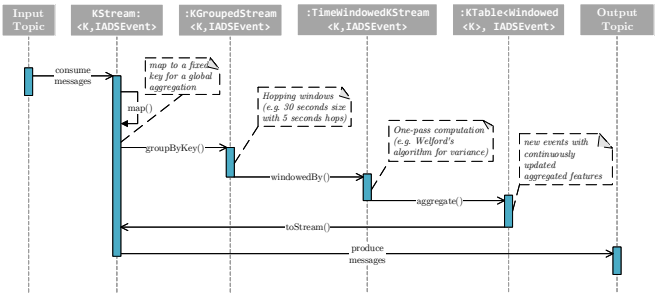

Fig. 3. Example of a feature aggregation by time windows using Kafka Streams DSL API [3].

Moreover, Apache Kafka fits the idea of supporting different types of message priorities and service goals such as throughput, latency, durability or availability [31]. In that sense, a set of practical experiments was conducted using Kafka to explore and better understand how its settings can be used to tune the platform to meet the requirements for next-generation IACS. Overall, the experiments show that the design of the event streaming layer, based on Apache Kafka, is not only flexible enough to meet different deployment scenarios but also able to be used as an efficient messaging broker mechanism capable of coping with the large number of events expected on more complex IACS environments.

On the other hand, the proposed data analytics concept serves as an aggregation point for all the events originating from multiple domains. It serves as an elastic approach where multiple computation nodes can cooperatively and dynamically perform a set of tasks. These processing tasks, which can instantiate both slow or fast processing mechanisms, may, for instance, implement multiple ML-based anomaly detection algorithms. Instead of using popular frameworks such as TensorFlow [32] or Scikit-learn [33] – which are mainly focused on ML – Apache Spark [34] was selected as the underlying computation platform. Spark is a general-purpose computation framework with native support of both streaming and batch processing, which provides additional flexibility to implement different types of intrusion and anomaly detection.

Besides experimental work focused on assessing the system scalability, the entire process of detecting a data exfiltration operation – a real threat for IACS environments – was used to demonstrate the proposed approach, including a proof of concept implementation and the integration and evaluation of various ML-based algorithms. Figure 4 illustrates the setup used to recreate a data exfiltration scenario using Domain Name System (DNS) tunnels. On the attacker's side, a server was setup in the cloud to behave as an authoritative DNS server for two previously registered domains. Then, two popular DNS tunneling tools (dnscat2 [35] and iodine [36]) were used to produce 23 variations, including simple tunnel handshakes using different DNS record types to encrypted sessions, interactive shells, and the exfiltration of a complete PLC project.

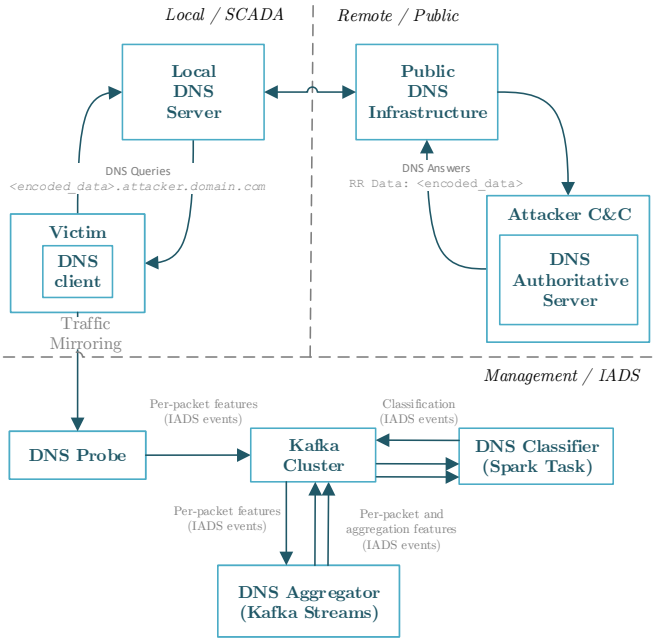

Fig. 4. Data exfiltration Scenario [20]

On the detection side, to showcase the different anomaly detection capabilities, a DNS probe was used for extracting packet-specific features (as described below) from the DNS network traffic. Then, a domain processor capable of extracting additional features, based on the aggregation of the messages into time windows, was created. Finally, an anomaly detection pipeline was created for training/classifying DNS traffic as normal or abnormal (i.e. a possible attempt of DNS tunneling).

Table I summarises the obtained results. This use case highlights the benefits of the analytics layer to detect complex threats and demonstrates the flexibility of the proposed approach in combining different algorithms. Regarding the proof-of-concept implementation, Apache Spark has proven to be a good match for developing such an analytics layer, enabling efficient and distributed computation capabilities. Besides performance, the Spark-rich feature set and its APIs for third-party integration have also proven to be a good choice, supporting the idea of a unified approach for streaming and batch processing.

V. Conclusions

We started by exploring several practical experiments and attack scenarios using Modbus and PCOM. The lack of previous literature about the security of PCOM motivated a more ambitious analysis, from both the attacker's and the defendant's points of view, that show how less known SCADA protocols can be used to conduct attacks in IACS. This work eventually led to several contributions to open-source tools such as Wireshark, Snort, Metasploit, Nmap and Scapy.

Next, we addressed the most significant perceived gap in the literature: the challenge of combining the vast array of data sources, probes and security components that characterise the next generation of IACS. To fulfil that gap, we leveraged recent advances in the fields of Big Data and event processing and assessed their suitability to IACS security management.

TABLE I
Summary of the key performance indicators using a dataset with aggregated network features [20].

Technique	Accuracy	Precision	Recall	F1	AUC
Decision Tree	0.9753	0.981	0.9699	0.9754	0.9753
Random Forests	0.9867	0.9887	0.985	0.9868	0.9867
GBT	0.981	0.9848	0.9774	0.9811	0.981
XGBoost	0.9886	0.9924	0.985	0.9887	0.9886
LightGBM	0.9734	0.9737	0.9737	0.9737	0.9734
Linear SVM	0.981	0.9923	0.9699	0.981	0.9811
MLPC	0.9867	0.9924	0.9812	0.9868	0.9868
Naive Bayes	0.9411	0.9916	0.891	0.9386	0.9416
Logistic Regression	0.981	0.9812	0.9812	0.9812	0.981

Area under ROC (UAC).

A data-driven holistic framework was proposed for monitoring the cyber-security of next-generation IACS and its two key components (streaming and data analytics) were presented and evaluated. This is a powerful way of decoupling data collection from data processing, domain-wise and location-wise.

In contrast to other works, we focused on combining the data and knowledge from different sources into a more comprehensive IADS approach. The advantage of having different types of specific security mechanisms at both local and global levels was a key driver for this work. Different ML-based mechanisms to detect anomalies in IACS environments were integrate and evaluated, highlighting the benefits of the proposed approach in terms of easily integrating additional anomaly detection techniques.

The complexity of modern IACS is still one of the biggest challenges for the anomaly detection process. The feasibility and practical evaluation of additional SCADA-specific algorithms proposed in the literature needs further work. Moreover, even though this thesis argues that SCADA security is not only about network communication protocols, most of the evaluation work focused on network-based scenarios – due to logistic and practical reasons. Nevertheless, it would be interesting to explore other types of data sources.

Reference Material

The work conducted in the scope of this thesis [3] directly led to several refereed publications [4], [5], [18]–[20], [37] and opensource contributions [9]–[17]. Moreover, it also contributed to several co-authored papers [6]–[8], [38]–[43].

Acknowledgment

This work was partially funded by the FCT–Foundation for Science and Technology, I.P./MCTES through National Funds (PIDDAC), within the scope of Centre for Informatics and Systems of the University of Coimbra (CISUC) Research and Development Unit, under Grant UIDB/00326/2020 and Project UIDP/00326/2020. It was also partially supported by the following projects: H2020 ATENA (H2020-DS-2015-1 Project 700581) and P2020 Smart5Grid (co-funded by FEDER -

Competitiveness and Internationalization Operational Program (COMPETE 2020), Portugal 2020 framework).

REFERENCES

[1] K. Stouffer, V. Pillitteri, S. Lightman, M. Abrams, and A. Hahn, "Nist special publication 800-82 revision 2. guide to industrial control systems (ics) security," *NIST*, 2015.

[2] International Electrotechnical Commission, "Industrial communication networks - network and system security - part 1-1: Terminology, concepts and models," 2018, https://www.isa.org/store/products/product-detail/?productId=116720.

[3] L. M. B. Rosa, "Intrusion and anomaly detection in industrial automation and control systems," Ph.D. dissertation, Universidade de Coimbra, 2022. [Online]. Available: http://hdl.handle.net/10316/101685

[4] L. Rosa, T. Cruz, P. Simões, E. Monteiro, and L. Lev, "Attacking SCADA systems: A practical perspective," in *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*. Lisbon, Portugal: IEEE, May 2017, http://ieeexplore.ieee.org/document/7987369/. [Online]. Available: http://ieeexplore.ieee.org/document/7987369/

[5] L. Rosa, M. Freitas, S. Mazo, E. Monteiro, T. Cruz, and P. Simões, "A comprehensive security analysis of a scada protocol: From osint to mitigation," *IEEE Access*, vol. 7, pp. 42 156–42 168, 2019, doi:10.1109/ACCESS.2019.2906926.

[6] T. Cruz, L. Rosa, J. Proença, L. Maglaras, M. Aubigny, L. Lev, J. Jiang, and P. Simões, "A Cybersecurity Detection Framework for Supervisory Control and Data Acquisition Systems," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 6, pp. 2236–2246, dec 2016, doi:10.1109/TII.2016.2599841.

[7] C. Foglietta, D. Masucci, C. Palazzo, R. Santini, S. Panzieri, L. Rosa, T. Cruz, and L. Lev, "From detecting cyber-attacks to mitigating risk within a hybrid environment," *IEEE Systems Journal*, vol. 13, no. 1, pp. 424–435, 2018, doi:10.1109/JSYST.2018.2824252.

[8] F. Adamsky, M. Aubigny, F. Battisti, M. Carli, F. Cimorelli, T. Cruz, A. Di Giorgio, C. Foglietta, A. Galli, A. Giuseppi *et al.*, "Integrated protection of industrial control systems from cyber-attacks: the atena approach," *International Journal of Critical Infrastructure Protection*, vol. 21, pp. 72–82, 2018, doi:10.1016/j.ijcip.2018.04.004.

[9] L. Rosa, "New snort rules for PCOM protocol," 2019, https://marc.info/?l=snort-sigs&m=154746968717558.

[10] ——, "pcomtcp: new built-in dissector for pcom protocol," 2019, https://code.wireshark.org/review/#/c/30823/.

[11] ——, "pcomtcp: dissection of additional PCOM/ASCII fields," 2019, https://code.wireshark.org/review/#/c/31467/.

[12] ——, "pcomtcp: PCOM/Binary command to descriptions," 2019, https://code.wireshark.org/review/#/c/31858/.

[13] ——, "Scapy - add a new PCOM layer," 2019, https://github.com/secdev/scapy/pull/1898.

[14] ——, "SCADA scan to collect information from unitronics plcs via pcom protocol," 2019, https://github.com/nmap/nmap/pull/1445.

[15] ——, "New module pcomclient," 2019, https://github.com/rapid7/metasploit-framework/pull/11219/.

[16] ——, "New PCOM module to send admin commands," 2019, https://github.com/rapid7/metasploit-framework/pull/11220/.

[17] ——, "PCOM pcap captures," 2019, https://github.com/lmrosa/pcom-misc/tree/master/pcaps.

[18] L. Rosa, J. Proença, J. Henriques, V. Graveto, T. Cruz, P. Simões, F. Caldeira, and E. Monteiro, "An evolved security architecture for distributed industrial automation and control systems," in *European Conference on Cyber Warfare and Security*. Academic Conferences International Limited, 2017, pp. 380–390.

[19] L. Rosa, M. Borges de Freitas, J. Henriques, P. Quitério, F. Caldeira, T. Cruz, and P. Simões, "Evolving the security paradigm for industrial iot environments," in *Cyber Security of Industrial Control Systems in the Future Internet Environment*. IGI Global, 2020, pp. 69–90.

[20] L. Rosa, T. Cruz, M. Freitas, P. Quitério, J. Henriques, F. Caldeira, E. Monteiro, and P. Simões, "Intrusion and anomaly detection for the next-generation of industrial automation and control systems," *Future Generation Computer Systems*, vol. 119, pp. 50–67, 2021.

[21] L. Rosa, "SCADA CIP enum scan," 2019, https://github.com/nmap/nmap/pull/1539.

[22] R. Leszczyna, *Cybersecurity in the Electricity Sector*. Springer, 2019.

[23] S. Nazir, S. Patel, and D. Patel, "Assessing and augmenting scada cyber security: A survey of techniques," *Computers & Security*, vol. 70, pp. 436–454, 2017.

[24] A. Humayed, J. Lin, F. Li, and B. Luo, "Cyber-physical systems security—a survey," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1802–1831, 2017.

[25] S. D. Antón, D. Fraunholz, C. Lipps, F. Pohl, M. Zimmermann, and H. D. Schotten, "Two decades of scada exploitation: A brief history," in *Application, Information and Network Security (AINS), 2017 IEEE Conference on*. IEEE, 2017, pp. 98–104.

[26] E. D. Knapp and J. T. Langill, *Industrial Network Security: Securing critical infrastructure networks for smart grid, SCADA, and other Industrial Control Systems*. Syngress, 2014.

[27] B. Feinstein, D. Curry, and H. Debar, "The Intrusion Detection Message Exchange Format (IDMEF)," RFC 4765, Mar. 2007, https://rfc-editor.org/rfc/rfc4765.txt. [Online]. Available: https://rfc-editor.org/rfc/rfc4765.txt

[28] Apache Software Foundation, "Apache kafka," 2020, https://kafka.apache.org/. [Online]. Available: https://kafka.apache.org/

[29] G. Hohpe and B. Woolf, *Enterprise integration patterns: Designing, building, and deploying messaging solutions*. Addison-Wesley Professional, 2004.

[30] Apache Software Foundation, "Apache kafka documentation," 2020, https://kafka.apache.org/documentation/. [Online]. Available: https://kafka.apache.org/documentation/

[31] Y. Byzek, "Optimizing your apache kafka deployment," 2019, https://cdn.confluent.io/wp-content/uploads/Optimizing_Your_Apache_Kafka_Deployment_White_Paper.pdf.

[32] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, https://www.tensorflow.org/. [Online]. Available: https://www.tensorflow.org/

[33] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[34] Apache Software Foundation, "Apache spark™ - unified analytics engine for big data," 2020, https://spark.apache.org/. [Online]. Available: https://spark.apache.org/

[35] R. Bowes, "dnscat2," 2019, https://github.com/iagox86/dnscat2. [Online]. Available: https://github.com/iagox86/dnscat2

[36] E. Ekman and B. Andersson, "iodine," 2019, https://github.com/yarrick/iodine. [Online]. Available: https://github.com/yarrick/iodine

[37] L. Rosa, P. Alves, T. Cruz, P. Simões, and E. Monteiro, "A comparative study of correlation engines for security event management," in *Iccws 2015-The Proceedings of the 10th International Conference on Cyber Warfare and Security*, 2015, p. 277.

[38] A. Lima, L. Rosa, T. Cruz, and P. Simões, "A security monitoring framework for mobile devices," *Electronics*, vol. 9, no. 8, p. 1197, 2020, doi:10.3390/electronics9081197.

[39] V. Graveto, L. Rosa, T. Cruz, and P. Simões, "A stealth monitoring mechanism for cyber-physical systems," *International Journal of Critical Infrastructure Protection*, vol. 24, pp. 126–143, 2019, doi:10.1016/j.ijcip.2018.10.006.

[40] B. Stewart, L. Rosa, L. A. Maglaras, T. J. Cruz, M. A. Ferrag, P. Simoes, and H. Janicke, "A novel intrusion detection mechanism for scada systems which automatically adapts to network topology changes," *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, vol. 4, no. 10, 2017, doi:eai.1-2-2017.152155.

[41] M. Borges de Freitas, L. Rosa, T. Cruz, and P. Simões, "Sdn-enabled virtual data diode," in *Katsikas S. et al. (eds) Computer Security. SECPRE 2018, CyberICPS 2018. Lecture Notes in Computer Science, vol 11387*. Springer, Cham, 2018, pp. 102–118.

[42] ——, "Sdn-enabled virtual data diode," in *Computer Security*. Springer, 2018, pp. 102–118.

[43] B. Stewart, L. Rosa, L. Maglaras, T. J. Cruz, P. Simões, and H. Janicke, "Effect of network architecture changes on ocsvm based intrusion detection system," in *International Conference on Industrial Networks and Intelligent Systems*. Springer, 2016, pp. 90–100.