

Universität Regensburg
Fakultät für Wirtschaftswissenschaften
Lehrstuhl für Wirtschaftsinformatik I - Informationssysteme

Collaborative Security with Cyber Threat Intelligence



Dissertation

zur Erlangung des Grades eines Doktors der Wirtschaftswissenschaft
eingereicht an der Fakultät für Wirtschaftswissenschaften der Universität Regensburg

vorgelegt von:

Daniel Schlette, M.Sc. with Honors

Berichterstatter:

Prof. Dr. Günther Pernul

Prof. Dr. Bernd Heinrich

Tag der Disputation: 26.07.2023

To my parents, Beate and Joachim.

Acknowledgement

University is this place where knowledge is accumulated by bringing people together. Intended to propel both individuals and society, universities encourage collaboration. Collaborative efforts are captured two-fold by my cumulative dissertation. First, they are ingrained in the research topic itself. However, joint work and support also shaped my journey throughout the last four years, which this acknowledgement aims to recognize.

First, collaborative cybersecurity has only been possible with the support of my supervisor Prof. Dr. Günther Pernul, as his guidance helped me to establish a research vision and pursue it to the end. In the IFS seminar and beyond, he gave me the advice necessary to see the bigger picture around threat intelligence and inspired me to develop exciting research ideas. I would also like to thank my second supervisor Prof. Dr. Bernd Heinrich for the feedback on methodology and collaboration in the DEVISE project.

Second, collaborative cybersecurity within research projects needs to be acknowledged. The works in this dissertation often resulted from fruitful discussions with Marco Caselli and his very valuable two cents on almost every topic. I am deeply grateful for his trust in me and the industry insights from Siemens, which enriched my research. I would also like to thank Thomas Schreck for making this collaboration possible and the German ministry BMBF funding the DEVISE project.

Third, collaborative cybersecurity with colleagues of the IFS team has defined my research and workday. On an administrative and technical side, Petra and Werner were a steady source of kindness and database support. Special thanks are due to Philip and Lena, who at the office, uni sports, and beyond assisted me with companionship, motivation, and their passion for research, as numerous improved papers show. I thank Bene for sharing the office, blockchain wisdom, and ambitions in sports. Collaboration with Manfred, Fabian, and Florian helped me establish a foothold at the chair with master thesis, initial projects, and CTI sharing. Having observed a generation shift, thanks go out to all my other colleagues: Marietheres, Ludwig, Sabrina, Thomas, Johannes, and Tobias. Despite the work focus, I also made new friends throughout these collaborations.

Finally, collaborative cybersecurity and this dissertation are based on the continuous support of friends and family. I am deeply grateful to my parents and my sister Isabel for their patience over the years, assistance in difficult situations, and running along. I owe the deepest gratitude to Vanessa for being at my side through all the ups and downs in the last four years and understanding everything. Thank you for all the love, laughter, and support you gave me.

Abstract

Frequent data breaches and security incidents show that organizations face challenges in protecting their information systems. To succeed, adversaries exploit vulnerabilities and launch targeted attacks, whereas defenders must constantly be on guard coping with increasingly complex systems. In this diametric situation, collaborative security facilitated by Cyber Threat Intelligence (CTI) seeks to level the playing field by promoting shared security information and data-driven approaches. When creating, sharing, and using threat intelligence, organizations can achieve more together. Nevertheless, technical and organizational problems still require research attention. In conducting information systems research, this dissertation contributes to three domains. First, it discusses standards and data formats indicating a shift toward actionable, procedural guidance with structured incident response representations. Second, it emphasizes the importance of sufficient data quality suggesting measures to assess and improve threat intelligence quality. And third, it explores novel incident response playbooks finding that playbooks assist organizations' security operations and are shaped by influencing organizational factors. The progress made in collaborative security holds promise for a more secure future.

Contents

Acknowledgement	i
Abstract	ii
List of Tables	v
List of Figures	vi
I Dissertation Outline	1
1 Motivation	2
2 Related Work	4
3 Research Questions	6
4 Methodology	7
4.1 Information Systems Research	7
4.2 Research Process	8
4.3 Research Setting	9
5 Results	10
5.1 Overview of Research Papers	10
5.2 Standards and Data Formats	11
5.3 Data Quality	13
5.4 Playbooks	17
5.5 Complementary Publications	23
6 Conclusion and Future Work	24
Bibliography	26
II Research Papers	30
1 A Comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective	31
2 Measuring and visualizing cyber threat intelligence quality	64
3 CTI-SOC2M2 – The quest for mature, intelligence-driven security operations and incident response capabilities	83
4 SOAR4IoT: Securing IoT Assets with Digital Twins	104
5 Generating ICS Vulnerability Playbooks with Open Standards	115

6	Do you Play It by the Books? A Study on Incident Response Playbooks and Influencing Factors	127
	Curriculum Vitae	146

List of Tables

1	Publications overview.	10
2	Complementary publications overview.	23

List of Figures

1	Simplified CTI life cycle.	3
2	Dissertation structure.	6
3	Mapping of research domains and publications.	11
4	CTI perspectives.	12
5	Structural incident response concepts.	12
6	Hierarchical structure of CTI quality dimensions.	14
7	Architecture of the CTI-SOC2M2 maturity model.	16
8	Prototypical implementation of the SOAR4IoT framework.	18
9	Process description for generating vulnerability playbooks.	20
10	Security advisory sources and classified workflow actions.	20
11	Playbook scenarios and the role of influencing factors.	22
12	Community playbooks characterized by the number of steps.	22

Part I

Dissertation Outline

1 Motivation

Collaborative cybersecurity is an inclusive concept incorporating different actors and information sharing. To combat threats, the cybersecurity community developed and exchanged recommendations and ideas on how to secure systems. These collaborative efforts led to the creation of handbooks and annual security conferences [37, 12]. Embedded into technology, security systems such as antivirus software build on collaboratively detecting computer viruses by sharing signatures [8]. Despite these and other approaches to secure information systems, we can observe today that the frequency, sophistication, and impact of attacks against organizations have changed [20]. Ransomware, mainly a threat to system and data availability but also to confidentiality, has evolved from a theoretical concept to a business model for cybercriminals [39, 24]. Attacker groups conduct supply chain attacks that leverage vulnerabilities in a globalized and software-based economy [7]. State-sponsored hacker groups amplify geopolitical tensions aiming for competitive advantages and upheaval within democratic societies [14]. However, defensive technology and the availability of data have evolved, too. Thus, it is paramount that security professionals across organizations embrace common philosophies such as “sharing is caring” to leverage existing resources to their full extent. For a better understanding and control of cyber threats, we need to collaborate more.

Our world today looks different because of the plethora of computing devices, digitization, and the accessibility of the internet. Digital products and services are more relevant to our societies than ever before [10]. This phenomenon is a leading cause of the threat landscape and the state of cybersecurity [20]. What can be attacked will eventually be attacked. Moreover, if there is more to gain, there is typically more incentive for malicious activity. On the contrary, securing information systems and fending off attacks becomes more demanding as attackers aim to succeed once while defenders must constantly be on guard. Threats, available data, and a drive toward collaborative security, often encouraged by regulation [9], lead organizations to focus on threat intelligence.

Cyber Threat Intelligence (CTI), a term used to refer to different types of security information broadly, has swiftly become emblematic of data-driven approaches in cybersecurity. Organizations are interested in CTI to complement security products (e.g., SIEM systems or SOAR tools) and teams. Moreover, organizations wish for a detailed understanding of threats and instructions on how to improve their cybersecurity [6]. Over the years, the cyber threat intelligence paradigm matured as researchers analyzed ways to structure and share security information [1, 30, 33, 26, 21, 4]. Nevertheless, theoretical and practical challenges remain and are addressed in this dissertation.

At its core, CTI bridges technical information with context about cyber threats. Therefore, CTI relies heavily on the basics of computers and networks. Most notably, IP addresses and malware file hashes are used as Indicators of Compromise (IoCs), supporting organizations in detecting threats [21]. On individual endpoints (e.g., servers, laptops), running processes, application log data, and software patch levels can indicate abnormal behavior or foreshadow security issues. What makes CTI go beyond pure

security information is context. Context is added through data analysis and conveys information about the victim and the attacker. Other organizations using the same technology (e.g., hardware or software) or operating within the same industry (e.g., military or healthcare) benefit if actionable CTI is shared [6]. Understanding attacker behavior and attack vectors can prevent future security incidents across organizations, as these attack characteristics are difficult to change and likely to be reused [3].

Combining a diverse set of security information in threat reports constitutes the strength of CTI. These threat reports hint at the origin of CTI rooted in military intelligence dossiers and incident reporting within organizations. Due to its influence on strategic and tactical decision-making in organizations, CTI is not exclusively technical [33]. From a process perspective, organizational CTI programs iterate the collection, analysis, and dissemination of threat intelligence. Borrowing from life cycle approaches, this thesis relies on a simplified threat intelligence life cycle depicted in Figure 1.

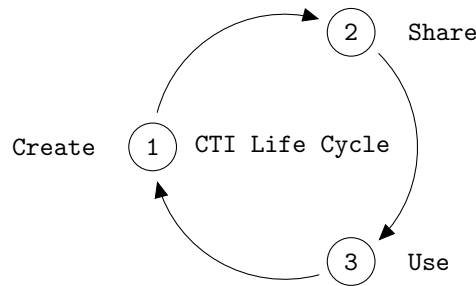


Figure 1: Simplified CTI life cycle.

Despite its benefits and practical relevance, organizations might be reluctant or overwhelmed to adopt CTI. Diverse information and different actors create challenges concerning standardization with data formats, data quality, and defensive measures with incident response playbooks. These challenges align with the above life cycle in which CTI is created, shared, and used, eventually leading to new insights. Structured representation with standards and data formats plays a role in each life cycle stage. While standards and data formats exist, keeping up with new developments is necessary to assist organizations. Building on standardization, CTI is only useful when it is timely, accurate, and complete. Sufficient data quality must be assured, yet little research has covered data quality assessment. At last, threat intelligence for security operations and incident response emphasizes process descriptions whereby novel playbooks constitute a means toward automation and other use.

To fill existing research gaps, the remainder of this dissertation is structured as follows. Section 2 discusses influential related work, providing relevant background information. Section 3 outlines the research questions and research domains to structure individual work items. Design science research methodology, research process, and research setting are part of Section 4. Section 5 details and maps the published papers contributing to the cumulative dissertation. Section 6 gives a conclusion and future work. This dissertation includes the original scientific publications, forming Part II.

2 Related Work

Research Development. Pinpointing the emergence of CTI is difficult as security information is tied to the use of computers and networks. However, the whitepaper by Sean Barnum and MITRE in 2014 about *Standardizing Cyber Threat Intelligence Information with the Structured Threat Information eXpression (STIX)* [1] marks a starting point. It used the term CTI and introduced STIX, a comprehensive, collaboratively developed data format for diverse security information. Since then, the number of scientific publications on the topic has grown. As of this writing (May 2023), the well-curated dblp computer science bibliography lists 266 publications for the search term “Cyber Threat Intelligence” and even more for the common abbreviation “threat intelligence”, which are used synonymously in this dissertation. Most notably, CTI research has been presented at high-ranked conferences such as the USENIX Security Symposium, ACSAC, ACM CCS, and ARES showing its relevance to security researchers.

Influential Research. CTI has been explored and shaped by different research efforts. If one has to read only a few academic papers, the following selection provides an overview of inspiring works that influenced this dissertation.

Building on security information, Mavroeidis and Bromander [23] outline and structure the components of CTI, while discussing their representation. CTI encompasses low-level indicators that allow us to understand Tactics, Techniques, and Procedures (TTPs) malicious actors use to achieve their strategic goals. Asset (i.e., hardware or software) identification with CPE, vulnerability identification and scoring with CVE, CWE, and CVSS, and various options for attack behavior classification (e.g., MITRE ATT&CK or CAPEC) are all considered part of CTI. Moreover, structured countermeasures help to cope with threats and security incidents, as CTI can be used for attack prevention, detection, and response.

Standardization with data formats is of particular importance for information sharing. Skopik et al. [30] discuss sharing security information as the backbone of collaborative cybersecurity. Mainly two data formats – STIX and MISP – enable the comprehensive representation and sharing of CTI. In comparison, STIX provides multiple distinct description elements (e.g., IPv6 address object or Malware object) [28], whereas MISP centers on event-attribute-tag combinations and is widely used due to its sharing platform [35, 32]; otherwise, practitioners also use the VERIS framework [34]. Tounsi and Rais [33] add to the CTI research by categorizing strategic, operational, tactical, and technical information. Further, they identify standardization and data quality as challenges around evidence-based knowledge for informed decisions across organizations.

Using and sharing CTI relies on adequate data quality. Li et al. [21] investigate collected IP address threat intelligence and malware file hash threat intelligence from a plethora of data feeds. The authors develop intuitive metrics (e.g., “accuracy” as false positives and “coverage”) for their data assessment and draw conclusions on the limited threat intelligence quality. They point out that most IP addresses are unique across

different data feeds, and standardized data labeling needs to be included as existing labels (e.g., “malicious” or “suspicious”) do not reveal any context information. Similarly, Bouwman et al. [4] conduct research on open-source and paid threat intelligence, assessing overlap and timeliness of indicators. With little overlap between vendors’ CTI, the authors additionally find that in the absence of a ground truth, organizations value fewer data in light of constraints on analyst time. In the data context, there is also a rich stream of research on mining CTI from public data sources [22, 13, 40, 19]. Typically, threat reports created and published by cybersecurity vendors combine diverse CTI and are thus pivotal for CTI mining and threat understanding.

At last, the necessity to establish organizational processes and procedures dealing with security risks is interwoven with CTI. In particular, when incidents happen or changes to the threat landscape occur, step-by-step guidance for security operations is pivotal. Stevens et al. [31] explore incident response playbooks that help organizations to handle security incidents and ongoing attacks. The authors focus on frameworks for playbook design and evaluate playbook effectiveness with a two-fold user study. Beyond academia, playbooks received additional attention in the US President’s Executive Order 14028: *Improving the Nation’s Cybersecurity*, urging governmental institutions to develop playbooks. Thereby, playbooks are defined as “a standard set of operational procedures to be used in planning and conducting cybersecurity activity” [11].

Industry Developments. CTI pertains to organizations and their cybersecurity [6]. Thus, it is a topic actively developed by corporate enterprises using CTI and vendors providing cybersecurity tools and services. The standardization of threat intelligence with data formats (e.g., STIX and MISP) has been shaped by participation from major corporations, institutions, and dedicated individuals [28, 35]. Moreover, in recent years the market of CTI products and services (e.g., sharing platforms) has seen steady growth, showing the relevance of CTI [18, 38]. While CTI matured due to industry support, accompanying research efforts enabled the delimitation of the CTI paradigm from other cybersecurity domains. In this dissertation, the collaboration with industry is leveraged to derive relevant problems and consider industry perspectives. Consequently, Information Systems (IS) research provides a suitable research methodology.

Against the backdrop of the aforementioned influential research and industry developments, this dissertation aims to add further insights into threat intelligence. Due to the continuous developments of standards and data formats, this dissertation builds on existing data formats while exploring shifts and recent approaches. Selecting the popular STIX standard, data quality and its assessment are approached in consideration of data quality management. Focusing on incident response playbooks, standards and data quality are recurring topics to understand how this novel CTI domain can help organizations to improve their cybersecurity. Use cases for CTI are manifold, but the research of this dissertation aims to pave the way for better documentation, automation, and in general, the use of threat intelligence for collaborative cybersecurity.

3 Research Questions

Threat intelligence continues to evolve. Hence, new research possibilities open up while other problems have already been identified by existing research and are promising to explore. Based on individual research gaps, this cumulative dissertation targets one overarching research question on threat intelligence.

Main research question.

How can organizations create, share, and use Cyber Threat Intelligence?

The main research question covers a broad area of problems on threat intelligence and organizational involvement throughout its life cycle. To better grasp the problem space concerning CTI and collaborative cybersecurity, more specific research questions focus on three CTI domains: 1) Standards and Data Formats, 2) Data Quality, and 3) Playbooks. The dissertation structure with the CTI domains is shown in Figure 2.

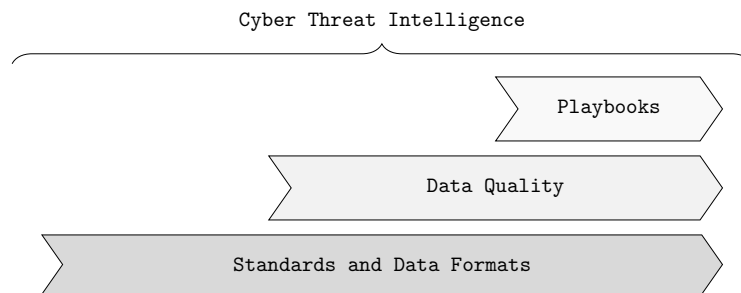


Figure 2: Dissertation structure.

Here, it is essential to discuss the rationale of the dissertation structure and how it relates to the simplified cyber threat intelligence life cycle (see Figure 1). With most and early research on CTI addressing **standards and data formats**, these structuring elements build an underlying foundation for other more specialized CTI research domains. At the outset of CTI creation, organizations face the selection of standards and data formats which also assist the sharing and use of CTI later on. Understanding the possibilities and limitations of standards and data formats allows us to focus on the more specific problems around adequate **data quality**, which surfaced over time. Organizations must consider which information is included in (semi-)structured data formats and their expectations on shared CTI. Without sufficient data quality, threat intelligence is of little value to organizations despite the use of data formats. **Playbooks** are represented with data formats and rely on data quality. Documenting a recent shift toward actionable CTI, playbooks embody CTI use as they capture processes and procedures.

Consequently, the main research question is split into three specific research questions. Research questions **RQ1**, **RQ2**, and **RQ3** each cover one of the three CTI domains outlined before. Furthermore, scientific publications forming the cumulative dissertation pick up on these research questions to structure and disseminate the results.

RQ1: Standards and Data Formats.

How can standards and data formats structure threat intelligence?

In the context of standards and data formats, the problem at hand is the continuous development (i.e., new versions, new approaches, shifts in perspective) and suitability for different organizational use cases. Thus, the solution approach consists of an in-depth analysis of the status quo capturing the most recent trends. Combining a structured literature search and observations of community efforts aims to paint a comprehensive picture that includes elements for comparison.

RQ2: Data Quality.

How can data quality assessment guide organizational threat intelligence?

Data quality assessment is necessary to improve threat intelligence quality and ensure effectiveness, but it has yet to be thoroughly explored. The solution approach to address this problem builds on existing works on data quality, including data quality dimensions and metrics. At the organizational level, the development of maturity models follows established guidelines.

RQ3: Playbooks.

How can incident response playbooks assist organizational cybersecurity efforts?

Incident response playbooks, offering step-by-step guidance for cybersecurity actions, have emerged as a novel topic of interest to both practitioners and researchers. However, a detailed examination of these playbooks is currently missing. To extract new findings and aid the understanding of playbooks, fields of application (i.e., Internet of Things, IoT), vulnerability management, and conceptual challenges of organizational playbook use are targeted by the solution approach. The applied research methods mainly center on data collection and prototyping.

4 Methodology

To ensure reliability, scientific research must be conducted systematically. The outlined research questions and proposed individual solutions are subsumed under information systems research. Accordingly, the research process follows the established design science research methodology and incorporates more specific methodologies as needed.

4.1 Information Systems Research

The research contributing toward this dissertation has been conducted at the University of Regensburg, Germany and is linked to the academic discipline of *Wirtschaftsinformatik*, which in essence refers to information systems research.

According to Hevner et al. [16], information systems research describes problem-driven approaches bridging practical relevance and scientific rigor. Two complementary and inseparable research paradigms define information systems research: behavioral science and design science. Behavioral science is centered on theories to explain human and organizational behavior. Design science is a research approach that focuses on designing and evaluating new artifacts. In both cases, the initiating condition for the research is a problem derived from a given environment consisting of people, organizations, and technology, also commonly understood as three cornerstones of cybersecurity.

This dissertation follows mainly the design science paradigm focusing on the design of artifacts, such as constructs, models, methods, and instantiations (e.g., prototypes). The design science research is complemented by behavioral science to emphasize organizational aspects and theoretical foundations. While theory building is typically associated with natural science, it can be argued that design science research itself encompasses design theory describing and communicating “how to do something” [15, 17]. Rooted in design science, the dissertation structure (see Figure 2) serves as a construct explaining the topic and challenges of CTI research. Individual publications follow the design science research methodology [29].

4.2 Research Process

Defined by Peffers et al. [29], Design Science Research Methodology (DSRM) includes the following iterative research process applied to the research questions on CTI.

Problem identification. Specific scientific or practical problems in the three CTI domains are identified and justified in each publication of this dissertation.

Objective definition. Artifact objectives are defined. RQ1 aims at structuring CTI and novel perspectives. RQ2 aims at measuring data quality and organizational integration. RQ3 aims at defining and understanding playbooks.

Design & Development. Artifacts are created considering artifact objectives and other requirements. Design and development are supported by systematic approaches such as literature reviews [36], maturity model [2] and taxonomy development [27], thematic analysis [5], and software development practices.

Demonstration. Use and suitability of the developed artifacts to solve the stated problems are demonstrated via their application.

Evaluation. Artifacts are evaluated qualitatively and quantitatively, determining how well objectives are reached. Semi-structured interviews, technical comparisons, performance assessments, and empirical evidence are used for the evaluation.

Communication. Research is presented to relevant audiences. Publishing scholarly research in scientific venues is immanent to the cumulative dissertation. When artifacts are implemented, source code is made publicly available.

4.3 Research Setting

This dissertation is embedded in a research setting consisting of previous works at the chair and two externally funded research projects. While the previous research has been used as an inspiration, the research projects have been used to collaborate with project partners and to communicate the research results. For a better understanding of the results of this dissertation, the following includes a brief description of the two aspects.

Onboarding. CTI research has been conducted at the chair before within another PhD project. Therefore, this dissertation is a successor to a previous dissertation on CTI sharing [25]. In its core publication, Menges and Pernul [26] show that CTI has matured, and standards and data formats are crucial to foster its sharing. While still a completely novel research topic back then, this dissertation builds on this previous research in multiple ways. First, it considers changes to CTI standards and data formats and their usage. Second, it emphasizes additional challenges around data quality only partially addressed in the previous PhD project via Distributed Ledger Technology (DLT). Third, it sets out to novel data-driven aspects with incident response playbooks targeting the use of CTI. Like the prior dissertation, the methodology in this dissertation is centered on design science research and includes a comparative analysis of CTI formats.

Industry Research Project. Research projects have accompanied and enabled this dissertation from the beginning. Initially, an industry research project with Siemens AG has been the key driver of both this dissertation and its topic. Throughout the project's work packages, this dissertation shaped and valuable industry insights were combined with a sound scientific basis derived from other research in the field. While aiming for both project success and dissertation progress, the methodology in this dissertation was influenced by directly capturing industrial ideas, feedback, and validation. This dissertation contributed to the successful extension and completion of the project two-fold. Besides the work package output, the dissertation yielded three scientific publications due to the collaboration. Additionally, results have been presented within Siemens and at the Forum of Incident Response and Security Teams (FIRST) conference 2022 in Dublin, a major annual industry conference.

Government Research Project. Funding for this dissertation was provided by the German Federal Ministry of Education and Research as part of the BMBF DEVISE project. The DEVISE project's goal is to develop a maturity model for data quality in the domains of CTI and Identity and Access Management (IAM). Therefore, this dissertation has taken an angle to explore CTI quality challenges. Collaboration with project partners from other security backgrounds and information system directions influenced research ideas and methodology. It often provided an accurate reality check. Consequently, two publications directly address CTI quality. One publication targets maturity model development specifically, and the others include elements related to data quality.

5 Results

A brief overview of the research papers documents the results and maps publications to the three key research areas: standards and data formats, data quality, and playbooks. Structured accordingly, a detailed summary of each research paper highlights addressed problems, applied methodology, key findings, and research implications. Each summary is concluded by emphasizing its contribution. Finally, the results close with complementary publications that provide additional insights into the research areas.

5.1 Overview of Research Papers

The cumulative research on CTI yielded six scientific publications, which are part of this dissertation. Table 1 provides a complete overview of each publication based on authors, title, and (planned) venue. For comparability and assessment of publication quality, the latest conference and journal rankings are included and refer to CORE ranking¹ and Impact Factor (IF), respectively.

Table 1: Publications overview.

No.	Publication	Ranking
P1	SCHLETTE, D., CASELLI, M., & PERNUL, G. (2021). A Comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective. <i>IEEE Communications Surveys & Tutorials</i> , 23(4), pp. 2525-2556.	25.249 IF
P2	SCHLETTE, D., BÖHM, F., CASELLI, M., & PERNUL, G. (2021). Measuring and visualizing cyber threat intelligence quality. <i>International Journal of Information Security</i> , 20(1), pp. 21-38.	2.427 IF
P3	SCHLETTE, D., VIELBERTH, M., & PERNUL, G. (2021). CTI-SOC2M2–The quest for mature, intelligence-driven security operations and incident response capabilities. <i>Computers & Security</i> , 111, 102482, pp. 1-20.	5.105 IF
P4	EMPL, P., SCHLETTE, D., ZUPFER, D., & PERNUL, G. (2022). SOAR4IoT: Securing IoT Assets with Digital Twins. In <i>The 17th International Conference on Availability, Reliability and Security (ARES 2022)</i> , pp. 4:1-4:10. Best Paper Runner-Up.	B CORE
P5	EMPL, P., SCHLETTE, D., STÖGER, L., & PERNUL, G. (2023). Generating ICS Vulnerability Playbooks with Open Standards. Submitted to <i>The 18th International Conference on Availability, Reliability and Security (ARES 2023)</i> .	B CORE
P6	SCHLETTE, D., EMPL, P., CASELLI, M., SCHRECK, T., & PERNUL, G. (2024). Do you Play It by the Books? A Study on Incident Response Playbooks and Influencing Factors. Submitted to <i>The 45th IEEE Symposium on Security and Privacy (S&P 2024)</i> .	A* CORE

¹<http://portal.core.edu.au>

As previously discussed, this dissertation is structured into three CTI domains, also covering a chronological development of the research area from standards and data formats toward playbooks. Figure 3 maps publications P1 to P6 to these research domains. While all publications relate to standards and data formats, P1 provides a detailed examination and categorization. The topic of data quality is addressed by publications P2 and P3. Through publications P4 to P6 cybersecurity playbooks are focused.

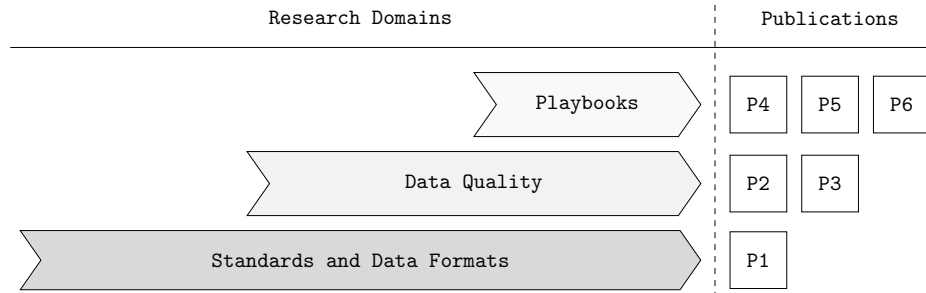


Figure 3: Mapping of research domains and publications.

5.2 Standards and Data Formats

To answer RQ1 describing how threat intelligence can be structured with standards and data formats, we conducted an extensive survey on recent developments in the field of CTI. Therefore, publication P1 addresses well-established standards and data formats but goes beyond existing literature by probing into different incident response representation efforts. The observations on threat intelligence representation indicate a shift in perspective broadening the current scope.

A Comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective [P1]

Joining forces and collaborating on cybersecurity requires a common understanding. Joint approaches and data formats help define agreed-upon objectives and representations of information. Recently, threat intelligence standards describing threats and security incidents (e.g., MITRE ATT&CK, OASIS STIX, or MISP) proliferated. However, beyond reporting on incidents and what attackers do, there is an emerging need to represent defensive procedures.

Several developments document the shift toward incident response representations displayed in Figure 4. Above all, there is a growing interest within the incident response community in using playbooks and courses of action to inform defenders what to do (e.g., OASIS CACAO or MITRE D3FEND). Besides, commercial solutions for *Security Orchestration, Automation and Response (SOAR)* address technical elements of incident response, advancing simple ticketing systems. Moreover, the U.S. President's Executive Order 14028 (Improving the Nation's Cybersecurity) emphasizes the importance of standardizing processes with incident response playbooks.

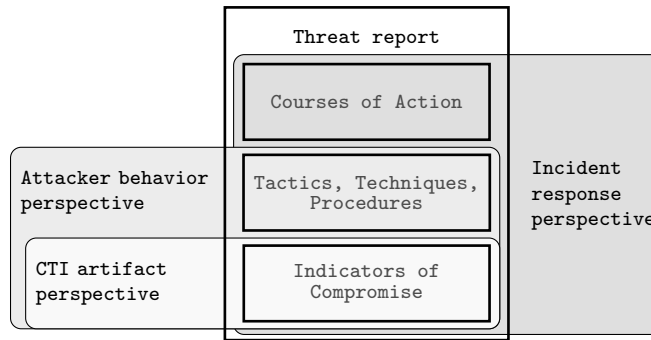


Figure 4: CTI perspectives.

Organizations searching for a structured representation of incident response, playbooks, or courses of action are confronted with different existing efforts. Therefore, guidance for comparing the various options and selecting the right standard for the right use case is crucial toward adoption and advancing collaborative cybersecurity.

In publication P1, we first assess and categorize the current state of threat intelligence before systematically approaching new efforts. Our research is guided by the question: Which formats and representation options exist, and how can they be compared? Thereby, we combine Design Science Research Methodology (DSRM) with targeted literature reviews for individual formats and playbooks. With the overall aim of providing clarity and orientation, we analyzed the following incident response formats²:

- *Collaborative Automated Course of Action Operations (CACAO) for Cyber Security*
- *Collaborative Open Playbook Standard (COPS)*
- *Integrated Adaptive Cyber Defense (IACD) Framework*
- *Open Command and Control (OpenC2)*
- *RE&CT Framework*
- *Resilient Event Conditions Action System against Threats (RECAST) Framework*

As a prerequisite for analysis, we first determine incident response formats' key aspects (i.e., core concepts). We structure 18 core concepts into four categories (i.e., general, structural, technological, or security). For structural concepts outlined in Figure 5, we observe that formats encompass a workflow in which individual steps are typically composed of an actuator performing an action on a given artifact.

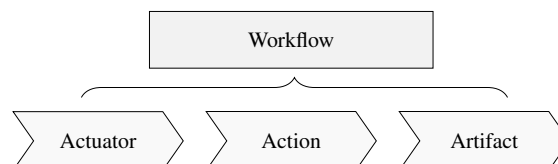


Figure 5: Structural incident response concepts.

²MITRE D3FEND is another promising framework for cybersecurity countermeasures as of this writing.

Our analysis using all core concepts yields a comparative summary across formats. As expected, there is a consistent focus on incident response actions. However, as formats have specific objectives, concepts are implemented differently or not at all. The analysis further reveals that organizations might experience difficulties finding adequate information about formats which can impair adoption. Furthermore, we observe the importance of both playbooks and frameworks for incident response representation.

Finally, several use cases exist where structured incident response representations can be of value. We suggest using the core concepts to filter available formats. For instance, in the case of automating incident response, structural and technological concepts are mandatory, thus indicating the use of OpenC2. In contrast, sharing incident response playbooks mandates “aggregability”, and CACAO becomes relevant. For reporting incident response capabilities, “categorization” or “readability” is mandatory, and the RE&CT framework is a viable option. Nevertheless, organizations can choose custom weights to determine formats for their specific use case.

We believe our research results can be a stepping stone for organizations to better understand and use structured representations for incident response. As new formats emerge, our analysis can be extended. In the context of DSRM and communication, the results of publication P1 have been presented at the 34th annual FIRST conference 2022.

Contribution of P1: The paper documents a shift in threat intelligence perspective toward incident response representation with novel data formats. Using a systematic approach based on DSRM and literature review, we introduce criteria, particularly actuator, action, and artifact, to compare data formats. Our analysis allows organizations to understand format characteristics and drive the selection of incident response formats for automation, sharing, and reporting use cases.

5.3 Data Quality

To answer RQ2 describing how data quality for threat intelligence can be assessed and integrated into organizational management instruments, we developed a concept for CTI quality assessment and a CTI capability maturity model. Publication P2 addresses data quality, dimensions, and metrics typically used in other domains. Consequently, these elements are configured to fit the threat intelligence domain and its prominent CTI standard, forming a CTI quality concept. In publication P3, we bridge the gap between security operations centers and the use of CTI formats by developing a maturity model. The model’s capability levels refer to data quality. The results of both research papers emphasize the importance of data quality in CTI and can serve as a stepping stone for improvements in threat intelligence quality.

Measuring and visualizing cyber threat intelligence quality [P2]

Collaborative cybersecurity relies on shared threat intelligence. In the simplified CTI life cycle, the sharing phase follows the creation phase and builds on standards and data

formats. Whenever organizations receive high-quality threat intelligence, they can use this information to cope with security threats, ongoing attacks, and security incidents in various ways (e.g., to prevent, detect, remediate, or recover). However, extant research and security professionals indicate that inaccurate, incomplete, or outdated CTI is a major obstacle in organizational decision-making and security operations.

Data quality is a challenge in diverse contexts and has become more pressing with the advent of computers, data collection, and sharing possibilities. Tapping into the rich stream of data quality and data quality management research, we explore how to measure and visualize CTI quality. Security professionals working with threat intelligence and facing poor data quality are currently lacking dedicated data quality approaches for CTI. Therefore, it is important to conceptually develop CTI quality instruments addressing the need to ensure the quality of shared threat intelligence.

In publication P2, we follow DSRM to propose a concept for CTI quality assessment using the standardized CTI format STIX, specifically version STIX2. As a starting point, we observe the absence of established scientific data quality approaches in the CTI domain. Thus, our research is guided by the question: How to apply and configure data quality instruments for assessing CTI quality?

First, we determine relevant quality dimensions by analyzing data quality research and a few existing works that proposed initial collections of quality dimensions important for CTI. Through additional discussions with CTI researchers and practitioners, we derive a set of data quality dimensions for CTI shown in Figure 6. Here, it must be noted that these quality dimensions are proposed considering the CTI format STIX2. We have chosen STIX2 because it has been standardized by OASIS Open, made available to the public, and is supported by security professionals across different industries. STIX2 also has a number of inherent characteristics (e.g., threat reports, dedicated object types) that deem it suitable for quality assessment.

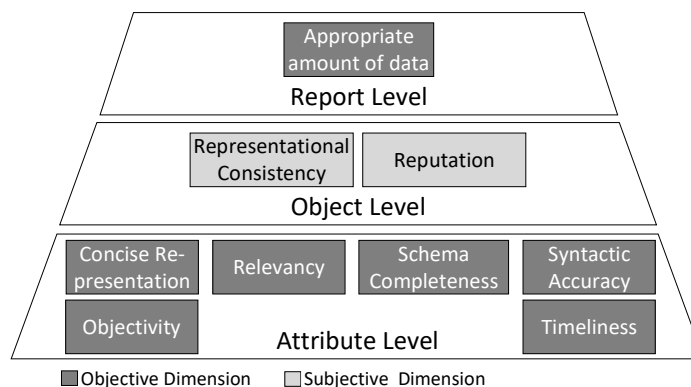


Figure 6: Hierarchical structure of CTI quality dimensions.

The quality dimensions for the STIX2 format are structured hierarchically. Dimensions can be quantified by evaluating unique attributes of STIX2 objects (attribute level). Other dimensions require additional external attributes or a holistic assessment of the given STIX2 object (object level). At last, the dimension “appropriate amount of data”

takes the complete threat description into account (report level). The hierarchical structure allows the aggregation of dimensions to quickly assess object quality and overall CTI report quality. We propose metrics with quantitative or structured qualitative (e.g., analyst perceptions) input for each dimension categorized into objective or subjective. To aggregate the quality dimensions into a single score for a given STIX2 report, we define an adjustable, weighted average across dimensions and hierarchy levels.

Beyond measuring CTI quality, our concept includes visualization elements relying on a previously developed prototype. Therefore, we extend the STIX2 data format to capture quality indicators. Defining a custom STIX2 quality object and data type, we persist quality information for the selected dimensions. In our prototype, we include an additional document-oriented database for performance reasons to store the JSON-structured quality indicators and link to the STIX object for which quality has been measured. We then adapt the visual display, integrating gauge visualization for the overall report and object quality, a quality tab for details on quality dimensions realized with progress bars and star ratings per object, and sliders to adjust dimensions weights.

In adherence with DSRM, we further evaluate the CTI quality concept encompassing dimensions, metrics, and visualization with three interviews. Interview participants confirm the high relevance of CTI quality assessment with the selected dimensions and metrics. While the amount of CTI has increased over time, organizations are searching for high-quality CTI. Our research results are promising as they can encourage future algorithmic implementation within threat intelligence sharing platforms.

Contribution of P2: The paper puts forth a concept to measure data quality for STIX2-structured CTI. Building on data quality research, we define, structure, and configure a relevant set of quality dimensions and metrics. Using visual representation, we make quality assessment transparent to analysts. Our approach is the first to include analysts' perceptions and is evaluated with interviews. Showing the relevance of CTI quality, we contribute to its assessment and integration in STIX2.

CTI-SOC2M2 – The quest for mature, intelligence-driven security operations and incident response capabilities [P3]

An organizational setting is implied when discussing collaborative security with cyber threat intelligence. Typically, the topic is handled within business units built to conduct and manage security and its operations. In the past, the terms *Computer Emergency and Response Team (CERT)* as well as *Computer Security Incident Response Team (CSIRT)* were frequently used to encapsulate security professionals working as security analysts, engineers, or architects. Emphasizing the operational aspects of security, nowadays, the *Security Operations Center (SOC)* fulfills a similar purpose by bundling security activities. However, from an organizational perspective it is vital to systematically assess the current state and determine improvement potential for the SOC.

Focusing on internal security information only has proven insufficient in a threat landscape that is constantly changing. While threat intelligence and its sharing proliferate, organizations must integrate this external information and build, assess, and improve security operations and incident response capabilities. Therefore, it is necessary to foster the organizational understanding and integration of CTI formats which can serve as a proxy for SOC service maturity.

In publication P3, we use methodologies for capability maturity model and taxonomy development to define the architecture and components of our model. Our model's objective is to assist the development of a mature, intelligence-driven SOC, which other models have not yet addressed. Initially, we ask: How can CTI elements be incorporated and structured within a capability maturity model?

First, we define the targeted problem and compare the envisioned model to existing maturity models. The resulting overview directs us to a development strategy and the iterative development of the CTI-SOC2M2 model. Our model has a three-tier architecture depicted in Figure 7 and is based on the CMMI framework. On the lowest tier, CTI formats function as an indicator and directive for capability fulfillment, eventually leading to maturity assessment. At the central tier, SOC services are grouped and constitute capabilities. The model specifies a capability level for each SOC service according to the use of CTI formats. On the upper tier, maturity levels indicate the current state of SOC maturity concerning CTI. Therefore we map CTI formats to SOC services and SOC services to the NIST Incident Response Life Cycle.

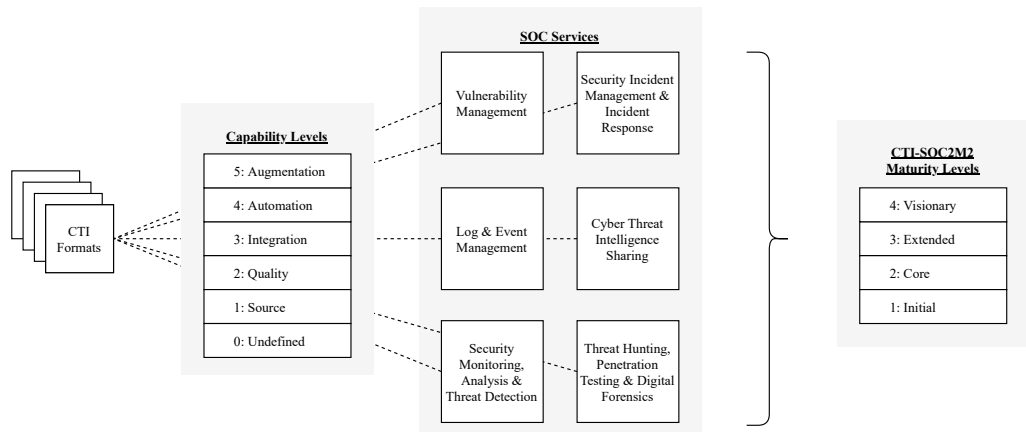


Figure 7: Architecture of the CTI-SOC2M2 maturity model.

As the developed model is data-driven, data quality aspects are embedded into the model defining the capability levels. For each SOC service, the following capability levels apply to the listed formats:

0: Undefined – CTI and CTI formats have not yet been considered.

1: Source – Have you determined and assessed the source of CTI with the mentioned CTI format(s)?

2: Quality – Have you applied appropriate measures to assess the quality of the CTI structured with the mentioned CTI format(s)?

3: Integration – Have you integrated CTI and the mentioned CTI format(s) into your organizational processes and technology architecture?

4: Automation – Have you automated retrieval, use and dissemination of CTI based on the mentioned CTI format(s)?

5: Augmentation – Have you set-up a monitoring mechanism to cope with new developments within CTI and new CTI format(s)?

We build on previous research defining SOC services to cover vulnerability management, log and event management, or incident management. Ultimately, maturity assessment is bottom-up, and reaching a higher maturity level implies higher capability levels and improved integration of CTI formats into the SOC. In our model, maturity levels refer to broader coverage and range from “Initial” to “Visionary”, indicating SOC maturity. We develop a web-based tool to allow organizations to quickly record their current state and draw conclusions on further improvements.

A two-fold evaluation underlines the importance of our model. In a quantitative user study, we observed a correlation between CTI formats and attack knowledge, inferring the maturity level of a SOC. Also, we conducted interviews which showed that the model is of practical relevance. Thus, our model can guide organizations toward adopting novel CTI formats and improving SOC services and CTI quality.

Contribution of P3: The paper develops an integrated capability maturity model for the use of CTI and its formats in a Security Operations Center (SOC). Following a mixed-methods approach, we derive SOC services and map CTI formats to assist organizations in determining their intelligence-based SOC maturity level. The model is implemented as a self-assessment tool and data quality constitutes a factor of organizational improvement.

5.4 Playbooks

To answer RQ3 describing how incident response playbooks can assist organizational cybersecurity efforts, we designed a framework combining Security Orchestration, Automation, and Response (SOAR) with the Internet of Things (IoT) in publication P4. In addition, we explore the connection between vulnerability management, its security advisories, and playbooks in publication P5. In publication P6, we address the foundations of playbooks in a comprehensive data-driven study. The results of these research publications contribute to the understanding and use of playbooks as they outline what playbooks are and which use cases they are suited for.

SOAR4IoT: Securing IoT Assets with Digital Twins [P4]

SOAR platforms provide functions to cope with security incidents and operational procedures. Typically, they integrate threat intelligence from external service providers and organizational assets to streamline security operations. Underlying SOAR platforms is

the need to handle the various security events generated by multiple security tools that might indicate a system breach or an ongoing attack. While commercial and open-source SOAR platforms exist, more research is needed to address integrating IoT assets and the development of playbooks for such purposes.

While a novel topic in itself, incident response playbooks and the IoT have yet to be explored together. In addition, the digital twin concept, allowing for device and system representation, has gained attention in the context of IoT devices and IoT networks. As IoT environments are characterized by multiple, heterogeneous devices and complex networks, they are vulnerable to attacks, too. Therefore, it is necessary to integrate them into a SOAR platform.

In publication P4, we design a SOAR4IoT framework using data abstraction with digital twins and playbooks to react to security events. We rely on DSRM and develop a prototype showing the framework's functioning. We are guided by the question: How to use Security Orchestration, Automation, and Response for the Internet of Things? More specifically, we aim to define SOAR, find ways to secure the IoT and implement SOAR for IoT with digital twins.

Our research first centers on SOAR platforms and the requirements to implement such platforms, for which we analyze literature and current SOAR platforms. Distilling a set of core activities and platform features, we assess whether these must be adapted to the IoT context. Toward the SOAR4IoT framework, we further analyze the incident response objectives of the IoT. We characterize different types of IoT attacks and determine their mitigation strategies. Complemented by a formal model, our SOAR4IoT framework has four components: IoT assets, middleware, SOAR platform with playbooks, and security tools. These components are connected, and the direction of the data flow is based on the SOAR core activity used. Using a proof of concept, we implement our framework as seen in Figure 8.

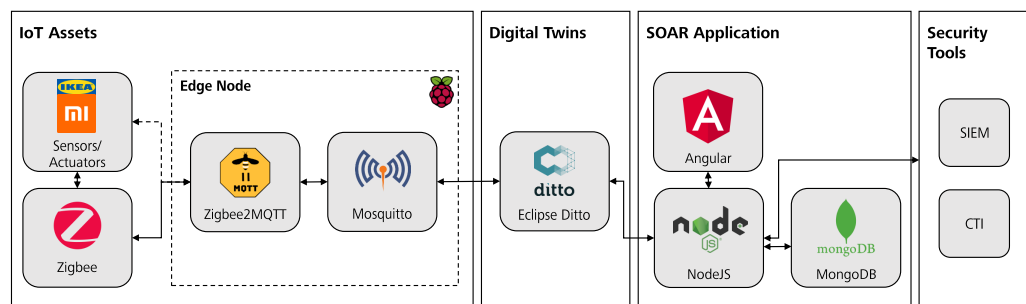


Figure 8: Prototypical implementation of the SOAR4IoT framework.

We set up a lab environment with different sensors and a dedicated edge node on a Raspberry Pi for the implementation. Our digital twins representing each IoT asset are realized with the Eclipse Ditto framework based on a JSON data structure. The SOAR application has an Angular frontend and NodeJS backend with MongoDB database and integrates microservices for the core app, digital twins, and security tools. Two use

cases are particularly important, a Mirai botnet and a Sybil attack, for which playbooks are developed to evaluate the prototypical implementation. Thus, our framework extends security operations to the IoT and leverages playbooks to do so. We add to the scientific discussion of playbooks in diverse fields of application. From a practical perspective, our open-source prototype can be adapted according to an organization's needs.

Contribution of P4: The paper introduces the SOAR4IoT framework bringing Security Orchestration, Automation, and incident Response (SOAR) and its playbooks to the Internet of Things (IoT). In an experimental setting, the framework is implemented using sensors, digital twins, and the MQTT protocol. In particular, we define and execute two playbooks for specific threats. Our results show that IoT asset management and security operations are feasible within the framework. The paper was awarded *Best Paper Runner-Up* at the ARES 2022 conference.

Generating ICS Vulnerability Playbooks with Open Standards [P5]

Security vulnerabilities are flaws in information systems based on human or computer deficiencies and predate security incidents. Product vendors aim to develop and sell secure products but are often unsuccessful, as numerous published vulnerabilities in the National Vulnerability Database (NVD) by the US National Institute of Standards and Technology (NIST) show. A steady stream of research targets vulnerabilities, their discovery, description with CVE IDs and text, assessment with CVSS scores, and patching. These activities of vulnerability management are important to organizations for secure operations. However, combining security advisories and playbook standards to generate vulnerability playbooks has yet to be explored.

Industrial Control Systems (ICS) enable the automation of production facilities in diverse industries. Unlike traditional IT environments, vulnerability patching in these environments has to cope with continuous operations, minimal downtime windows, and system complexity. As a consequence, patching is often deferred or not an option. For that reason, product vendors offer security advisories for vulnerabilities affecting their products which contain workarounds specifying other remediation measures. In addition, organizations wish for easy access to handle these remediation measures. Therefore, the development of vulnerability playbooks is a promising new option.

In publication P5, we develop a process model and generate ICS vulnerability playbooks. We use DSRM to develop a prototype based on the open standards CSAF and CACAO. We ask: Is it possible to generate ICS vulnerability playbooks? In addition, we are interested in classifying remediation measures (i.e., actions). Aiming to ensure smooth conversion of input security advisories in vulnerability playbooks, we evaluate our approach using confusion matrices for performance assessment.

At first, we develop a four-phase process model described in Figure 9. In the first and second phase, vulnerabilities for a given set of ICS assets are searched by CVE and device tags, and advisories are fetched. In the following phases, advisories are converted

to CSAF, if needed, remediations are extracted, and actions are classified with matching terms. Forming playbook steps, CACAO vulnerability playbooks are constructed.

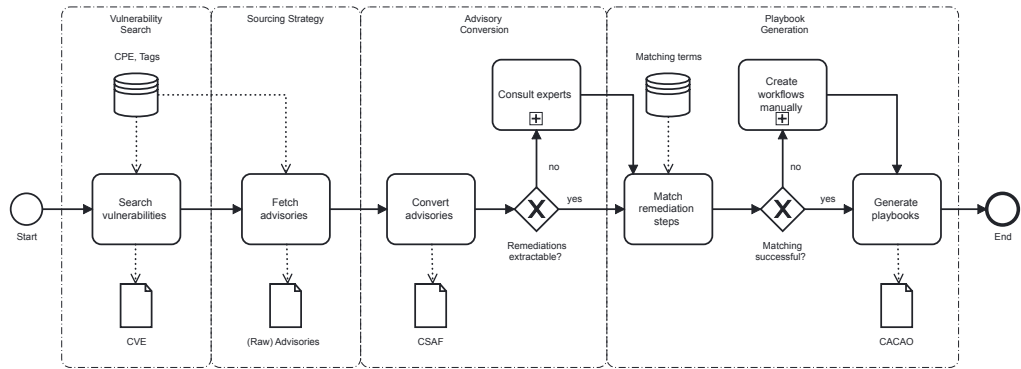
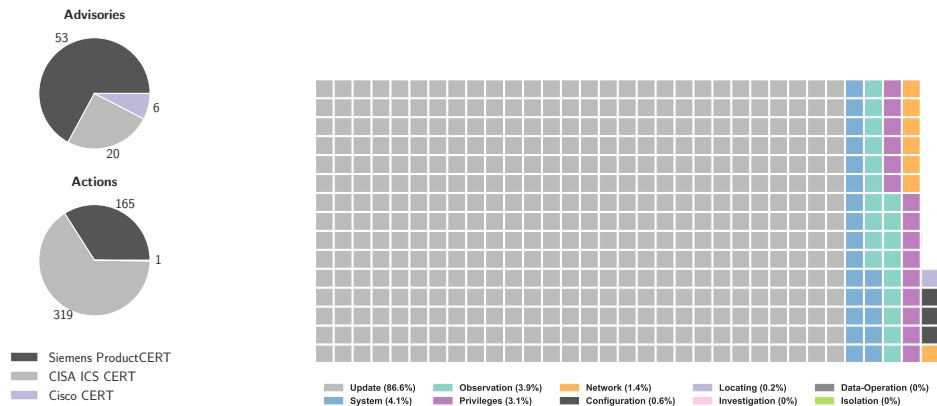


Figure 9: Process description for generating vulnerability playbooks.

Our approach includes a prototypical implementation with VueJS frontend and NodeJS backend. In addition, it uses a MongoDB database storing CSAF and CACAO documents structured in JSON. The application follows the model-view-controller principle and runs on a standard virtual machine with Ubuntu operating system. For the classification, OpenC2 action terms help to determine action classes. Using Natural Language Processing (NLP), we convert the remediation steps described by sentences into stemmed tokens for classification via term matching. In our application, user interaction is possible for selecting the matching terms and result validation. For CSAF advisories and CACAO playbooks, it has the complete standards implemented.



(a) Advisories and actions.

(b) Classification of 485 identified workflow actions.

Figure 10: Security advisory sources and classified workflow actions.

We extensively evaluate our approach relying on an industrial use case. Therefore, we run the application and query advisories from Siemens, Cisco, and a well-known governmental source, the US Cybersecurity & Infrastructure Security Agency (CISA). This input data includes 79 advisories and 485 actions visualized in Figure 10a. The evaluation then includes output in the form of CACAO playbooks, which we classified. Figure 10b shows that the 485 workflow actions cover predominantly update advice. Actions

that are part of the system class, observation class, and access class are observed less often. To evaluate performance, we manually label the correct detection of sentences with remediation measures and whether they have been transformed into CACAO playbook workflow actions using confusion matrices.

Our results show that vulnerability playbook generation is possible with adequate playbook quality and processing time. However, we see substantial benefits in improving security advisory quality and extending our work. Using the CSAF standard and making it easily available can reduce organizational hurdles to vulnerability management. Our work can provide initial guidance for additional automation and recommends mapping the playbooks' actions to platform-specific commands needed to fully implement vulnerability playbooks within organizations.

Contribution of P5: The paper proposes a systematic process for transforming vulnerability advisories into CACAO-structured playbooks for mitigation. Our prototypical application uses advisories from Industrial Control Systems (ICS) vendors as input, extracting and classifying relevant actions before generating playbooks. We evaluate our approach showing that actions are easier to identify for CSAF-structured advisories and document the potential of vulnerability playbooks.

Do you Play It by the Books? A Study on Incident Response Playbooks and Influencing Factors [P6]

Incident response playbooks are structured sets of operational procedures organizations use to instruct humans or machines on performing countermeasures against cybersecurity threats. Integral to SOAR platforms, they are offered by security vendors to advertise their products and adapted by organizations to their context. Despite the recent increase in the availability of playbooks, these have yet to be scientifically examined.

Organizations should define and document standard operating procedures to ensure consistent cybersecurity operations and incident response according to industry best practices. Playbooks representing such processes and procedures can fulfill various organizational use cases. For instance, organizations show interest in playbooks as they promise to streamline security operations, ideally leading toward the automation of services and the reduction of errors and tedious security tasks. While many organizations already use playbooks, sharing, maintaining, and using playbooks depends on understanding the playbook concept. Therefore, playbooks and organizational influencing factors are a topic worth being investigated.

In publication P6, we empirically research incident response playbooks. We conduct our research by asking: What is inside a playbook? On a more detailed level, this research question is specified: What are characteristics of community playbooks made available by trusted sources? Moreover, which influencing factors shape incident response processes and organization-specific playbooks?

Without an established theoretical foundation, we suppose that community playbooks contain generic, technical threat information but are stripped of any organizational context to be shared confidentially. Influencing factors relevant to a given organization add organizational aspects to community playbooks, making them organization-specific. Figure 11 describes playbook scenarios in which influencing factors play a role and how to generate organization-specific playbooks, keeping factors and playbooks separate.

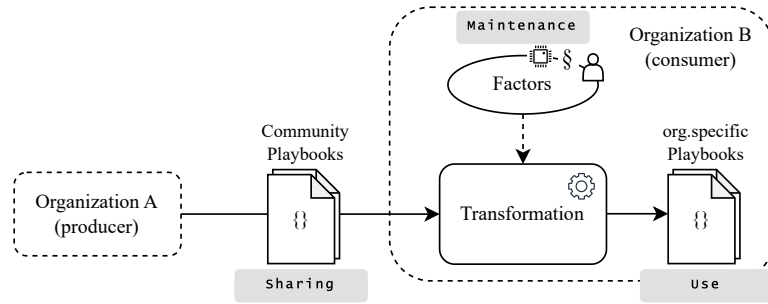


Figure 11: Playbook scenarios and the role of influencing factors.

First, we gather playbook data from SOAR vendors and other trusted sources. Figure 12 shows the 1221 playbooks queried from 14 sources and the median number of steps in these playbooks. While predominantly generic, community playbooks contain information about ticketing and tools likely handled differently across organizations. Our analysis using an online questionnaire with 147 participants extends these findings revealing that organizations use playbooks mainly for documentation, automation, compliance, and onboarding. Asking for factors influencing incident response and playbooks, we observe that technology, specific incident response directives, and people are mentioned most often. While the organizations that our participants work for have, on average, 24.5 playbooks, we notice a high standard deviation indicating stark organizational differences. Overall, we find that organizations have specific playbooks underlined by step count, parallel workflow, and influencing factors.

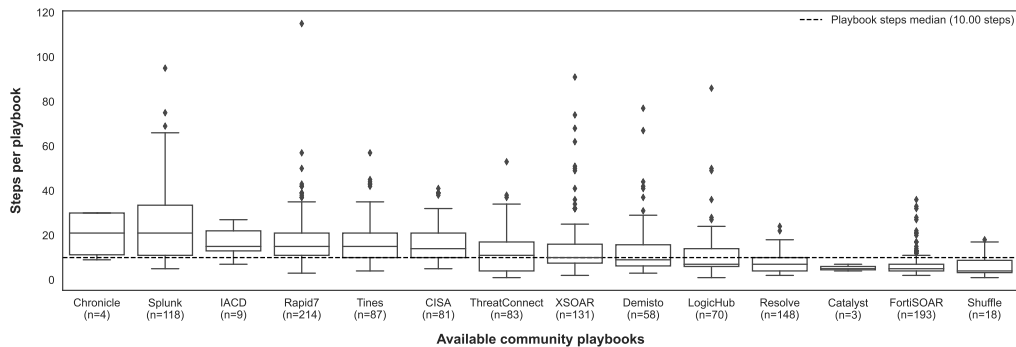


Figure 12: Community playbooks characterized by the number of steps.

A core contribution of this playbook study is achieved using interviews with security professionals. We conduct nine interviews to examine the implications of influencing factors on incident response processes and playbooks. We find that incident response

processes and playbooks are indeed shaped by influencing factors, but this is not necessarily recognized by practitioners. Statements by interview participants show that laws and regulations apply throughout, as emphasized by the following quote: “Only incident handlers in the US are allowed to handle incidents affecting US military contracts. We do reassign responsibilities accordingly.” We also find multiple examples of how other influencing factors shape processes and playbooks.

Our playbook study is the first to assess incident response playbooks empirically. Adding a theoretical foundation to the concept of playbooks, we aim to clarify the understanding and use of playbooks. While playbooks are well-known, understanding and use depend on implementation tools and abstraction levels. Thus, organizations must determine their intentions concerning flexibility and reuse. Influencing factors can be of value to organizations wishing to adapt playbooks to their context.

Contribution of P6: The paper is an in-depth study of incident response playbooks. Based on three data sources – community playbooks, online questionnaire, and interviews – the concept of playbooks is sharpened in light of sharing, maintaining, and using playbooks across organizations. Our analysis reveals playbook characteristics and that organizational factors influence playbooks. Organizations play it by the books, but intrinsic ambiguities define playbooks.

5.5 Complementary Publications

In addition to the main publications P1 to P6, this dissertation yielded complementary publications. Table 2 lists these additional works. Briefly summarized below, they contribute to a general understanding of CTI and its application for industrial environments.

Table 2: Complementary publications overview.

No.	Publication	Ranking
C1	SCHLETTE, D., MENGES, F., BAUMER, T., & PERNUL, G. (2020). Security Enumerations for Cyber-Physical Systems. In <i>Data and Applications Security and Privacy XXXIV: 34th Annual IFIP WG 11.3 Conference (DBSec 2020), Lecture Notes in Computer Science 12122</i> , pp. 64-76.	B CORE
C2	SCHLETTE, D. (2021). Cyber Threat Intelligence. In <i>Encyclopedia of Cryptography, Security and Privacy (3rd Edition)</i> , pp. 1-3.	N/A
C3	SCHLETTE, D. (2021). Cyber Threat Intelligence Sharing. In <i>Encyclopedia of Cryptography, Security and Privacy (3rd Edition)</i> , pp. 1-3.	N/A
C4	DIETZ, M., SCHLETTE, D., & PERNUL, G. (2022). Harnessing Digital Twin Security Simulations for systematic Cyber Threat Intelligence. In <i>46th Annual Computers, Software, and Applications Conference (COMPSAC 2022)</i> , pp. 789-797.	B CORE

In **Publication C1**, we explore the extension of two security enumerations, namely CPE and CVE, to capture the characteristics of Cyber-Physical Systems (CPS). We develop a conceptual meta-model and implement a prototype to show feasibility. The results were presented at the virtual *Data and Applications Security and Privacy (DBSec)* conference and served as a starting point for research on CTI standards and data formats.

Publications C2 and C3 are articles in the *Encyclopedia of Cryptography, Security and Privacy*. They concisely define the concepts of “Cyber Threat Intelligence” and “Cyber Threat Intelligence Sharing”. The encyclopedia articles resulted from a detailed topic examination but also show future research opportunities for CTI.

In **Publication C4**, we use simulation output by an industrial digital twin and the STIX standard to systematically generate threat intelligence. In essence, the paper is based on a process model and outlines steps on how to use the STIX standard in version 2.1. We can show that relevant network and log data is part of a digital twin simulation, but manual steps and missing context might constitute limitations. Useful CTI must be based on realistic simulation scenarios and consider context. Thus, this research helped to shape ideas of organizational and technical aspects of CTI.

6 Conclusion and Future Work

The overall objective of this dissertation was to explore collaborative security guided by the question: How can organizations create, share, and use Cyber Threat Intelligence? While simple answers always abstract reality, we can narrow down on aspects crucial to adopting threat intelligence. Essentially, this leads to the CTI domains which this dissertation addressed. Organizations can leverage the full potential of CTI when considering standards and data formats, assessing data quality, and understanding playbooks.

RQ1 has been addressed with a comprehensive analysis of CTI standards and data formats. The results indicate a shift toward incident response representations broadening the scope of CTI. Answering RQ1, the published work adds a detailed analysis to the literature. It further outlines the need to monitor and compare current developments. The security researchers’ perspective on standards also has implications for the industry. Organizations can use our work to select data formats fitting their use cases.

RQ2 has been addressed by developing instruments to assess and improve the data quality of CTI. The results are two concepts that help to tackle CTI quality problems. Answering RQ2, the published works transfer existing knowledge to CTI and align quality aspects. Consequently, practitioners can build on the assessment of CTI quality by developing specific algorithms and applying the maturity model in organizations.

RQ3 has been addressed by proposing prototypes and empirical research on playbooks. The results show that playbooks are suitable for various environments but are also influenced by organizational factors. Answering RQ3, this dissertation adds a theoretical foundation to the literature. Making intrinsic organizational ambiguities transparent assists organizations in understanding and adapting playbooks to their context.

Overall, this dissertation has contributed to the understanding of collaborative security with threat intelligence. It has captured existing challenges and proposed solutions. The individual works have been published following the peer-review process and are well received by the community, as indicated by citations. Research impact is also documented by funded research projects building on the results of this dissertation and aiming to explore CTI sharing with distributed ledger technology.

Throughout the work on this dissertation, CTI continued to mature. Arguably, there is still more collaboration among organizations and security professionals needed. Moreover, the research community can benefit from more innovative and groundbreaking research on threat intelligence. This dissertation tried to make an impact by exploring research questions in three CTI domains, but there are multiple opportunities to pursue additional research, and often the “simplest” research questions are the hardest to answer. In the following, two avenues for future work are discussed.

Usable CTI. Possibly the greatest challenge in CTI is assessing context and how to share *relevant* and understandable security information for different audiences. Therefore, it is worth considering the bigger picture and usable security. Understanding how CTI can assist in onboarding security analysts, propelling smaller organizations’ security, and defending major corporations necessitates research from an information systems background. Usable CTI is as much about technology (e.g., systems, commands) as it is about the audience (e.g., researchers, senior executives). It will be interesting to observe if novel distributed ledger technology can lower the hurdle to CTI sharing through incentive mechanisms. Aiming to develop robust security measures while retaining security professionals, future research must ensure appropriate CTI integration.

Large Language Models. Recently, significant advancements have been made in generative Artificial Intelligence (AI). Neural network-based Large Language Models have demonstrated the ability to produce relevant text and organize information effectively. While the “intelligence” in threat intelligence heavily relies on human analysis, tasks such as data correlation and structuring can benefit from generative AI. For instance, Microsoft’s Security Copilot already utilizes generative AI to generate threat reports and prompt books for security actions. Looking ahead, further exploration of generative AI in security contexts can uncover its evolving potential and influence on collaborative security and threat intelligence practices.

Closing remarks. This dissertation documents that collaborative security with threat intelligence is continuous work. However, research on standards and data formats, data quality, and playbooks also shows that the cybersecurity community is working together to overcome deficiencies. Driven by personal motivation, the right mindset, increasing organizational support, and regulation, the state of cybersecurity is not as grim as often purported. There is progress in collaborative security, and a future with more security rather than less awaits. It is on future research to continue the journey onward.

Bibliography

- [1] BARNUM, S. Standardizing cyber threat intelligence information with the structured threat information expression (STIX). Tech. rep., MITRE Corporation, 2014.
- [2] BECKER, J., KNACKSTEDT, R., AND PÖPPELBUSS, J. Developing maturity models for IT management. *Bus. Inf. Syst. Eng. 1*, 3 (2009), 213–222.
- [3] BIANCO, D. J. The pyramid of pain. <https://detect-respond.blogspot.com/2013/03/the-pyramid-of-pain.html>, 2013. Last accessed 2023-05-31.
- [4] BOUWMAN, X., GRIFFIOEN, H., EGBERS, J., DOERR, C., KLIEVINK, B., AND VAN EETEN, M. A different cup of TI? The added value of commercial threat intelligence. In *29th USENIX Security Symposium, USENIX Security 2020, August 12-14, 2020* (2020), S. Capkun and F. Roesner, Eds., USENIX Association, pp. 433–450.
- [5] BRAUN, V., AND CLARKE, V. Using thematic analysis in psychology. *Qualitative Research in Psychology 3*, 2 (2006), 77–101.
- [6] BROWN, R., AND STIRPARO, P. SANS 2022 cyber threat intelligence survey. Tech. rep., SANS, 2022.
- [7] CHESNEY, R. Solarwinds and the holiday bear campaign: A case study for the classroom. <https://www.lawfareblog.com/solarwinds-and-holiday-bear-campaign-case-study-classroom>, 2021. Last accessed 2023-05-31.
- [8] CISCO SYSTEMS, INC. ClamAV documentation. <https://www.clamav.net/>, 2023. Last accessed 2023-05-31.
- [9] EUROPEAN PARLIAMENT AND THE COUNCIL OF THE EUROPEAN UNION. NIS2 Directive (EU) 2022/2555 of the European Parliament and of the Council of 14 December 2022 on measures for a high common level of cybersecurity across the Union. <https://eur-lex.europa.eu/eli/dir/2022/2555/oj>, 2022. Last accessed 2023-05-31.
- [10] EUROSTAT. Digital economy and society statistics - households and individuals. https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Digital_

- economy_and_society_statistics_-_households_and_individuals, 2022. Last accessed 2023-05-31.
- [11] EXECUTIVE OFFICE OF THE PRESIDENT. Executive Order 14028 of May 12, 2021 – Improving the Nation’s Cybersecurity. <https://www.whitehouse.gov/briefing-room/presidential-actions/2021/05/12/executive-order-on-improving-the-nations-cybersecurity/>, 2021. Last accessed 2023-05-31.
- [12] FIRST – FORUM OF INCIDENT RESPONSE AND SECURITY TEAMS. Annual first conference on computer security incident handling. <https://www.first.org/conference/>, 2023. Last accessed 2023-05-31.
- [13] GASCON, H., GROBAUER, B., SCHRECK, T., RIST, L., ARP, D., AND RIECK, K. Mining attributed graphs for threat intelligence. In *Proceedings of the Seventh ACM Conference on Data and Application Security and Privacy, CODASPY 2017, Scottsdale, AZ, USA, March 22-24, 2017* (2017), G. Ahn, A. Pretschner, and G. Ghinita, Eds., ACM, pp. 15–22.
- [14] GOOGLE THREAT ANALYSIS GROUP. Fog of war: how the ukraine conflict transformed the cyber threat landscape. <https://blog.google/threat-analysis-group/fog-of-war-how-the-ukraine-conflict-transformed-the-cyber-threat-landscape/>, 2023. Last accessed 2023-05-31.
- [15] GREGOR, S. The nature of theory in information systems. *MIS Q.* 30, 3 (2006), 611–642.
- [16] HEVNER, A. R., MARCH, S. T., PARK, J., AND RAM, S. Design science in information systems research. *MIS Q.* 28, 1 (2004), 75–105.
- [17] JONES, D., AND GREGOR, S. The anatomy of a design theory. *J. Assoc. Inf. Syst.* 8, 5 (2007), 19.
- [18] LAWSON, C., BENSON, R., AND CONTU, R. Market guide for security threat intelligence products and services. Tech. rep., Gartner, 2021.
- [19] LEGOY, V., CASELLI, M., SEIFERT, C., AND PETER, A. Automated retrieval of ATT&CK tactics and techniques for cyber threat reports. *CoRR abs/2004.14322* (2020).
- [20] LELLA, I., TSEKMEZOGLOU, E., NAYDENOV, R. S., CIOBANU, C., MALATRAS, A., AND THEOCHARIDOU, M. ENISA threat landscape 2022. Tech. rep., European Union Agency for Network and Information Security (ENISA), 2022.
- [21] LI, V. G., DUNN, M., PEARCE, P., MCCOY, D., VOELKER, G. M., AND SAVAGE, S. Reading the tea leaves: A comparative analysis of threat intelligence. In *28th USENIX Security Symposium, USENIX Security 2019, Santa Clara, CA, USA*,

- August 14-16, 2019* (2019), N. Heninger and P. Traynor, Eds., USENIX Association, pp. 851–867.
- [22] LIAO, X., YUAN, K., WANG, X., LI, Z., XING, L., AND BEYAH, R. A. Acing the IOC game: Toward automatic discovery and analysis of open-source cyber threat intelligence. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016* (2016), E. R. Weippl, S. Katzenbeisser, C. Kruegel, A. C. Myers, and S. Halevi, Eds., ACM, pp. 755–766.
- [23] MAVROEIDIS, V., AND BROMANDER, S. Cyber threat intelligence model: An evaluation of taxonomies, sharing standards, and ontologies within cyber threat intelligence. In *European Intelligence and Security Informatics Conference, EISIC 2017, Athens, Greece, September 11-13, 2017* (2017), J. Brynielsson, Ed., IEEE Computer Society, pp. 91–98.
- [24] MELAND, P. H., BAYOUMY, Y. F. F., AND SINDRE, G. The ransomware-as-a-service economy within the darknet. *Comput. Secur.* 92 (2020), 101762.
- [25] MENGES, F. *Cyber Threat Intelligence Exchange*. PhD thesis, University of Regensburg, Germany, 2020.
- [26] MENGES, F., AND PERNUL, G. A comparative analysis of incident reporting formats. *Comput. Secur.* 73 (2018), 87–101.
- [27] NICKERSON, R. C., VARSHNEY, U., AND MUNTERMANN, J. A method for taxonomy development and its application in information systems. *Eur. J. Inf. Syst.* 22, 3 (2013), 336–359.
- [28] OASIS CYBER THREAT INTELLIGENCE (CTI) TECHNICAL COMMITTEE. STIX Version 2.1: OASIS Standard. Tech. rep., OASIS, 2021.
- [29] PEFFERS, K., TUUNANEN, T., ROTHENBERGER, M. A., AND CHATTERJEE, S. A design science research methodology for information systems research. *J. Manag. Inf. Syst.* 24, 3 (2007), 45–77.
- [30] SKOPIK, F., SETTANNI, G., AND FIEDLER, R. A problem shared is a problem halved: A survey on the dimensions of collective cyber defense through security information sharing. *Comput. Secur.* 60 (2016), 154–176.
- [31] STEVENS, R., VOTIPKA, D., DYKSTRA, J., TOMLINSON, F., QUARTARARO, E., AHERN, C., AND MAZUREK, M. L. How ready is your ready? assessing the usability of incident response playbook frameworks. In *CHI '22: CHI Conference on Human Factors in Computing Systems, New Orleans, LA, USA, 29 April 2022 - 5 May 2022* (2022), S. D. J. Barbosa, C. Lampe, C. Appert, D. A. Shamma, S. M. Drucker, J. R. Williamson, and K. Yatani, Eds., ACM, pp. 589:1–589:18.

- [32] STOJKOVSKI, B., LENZINI, G., KOENIG, V., AND RIVAS, S. What's in a cyber threat intelligence sharing platform?: A mixed-methods user experience investigation of MISP. In *ACSAC '21: Annual Computer Security Applications Conference, Virtual Event, USA, December 6 - 10, 2021* (2021), pp. 385–398.
- [33] TOUNSI, W., AND RAIS, H. A survey on technical threat intelligence in the age of sophisticated cyber attacks. *Comput. Secur.* 72 (2018), 212–233.
- [34] VERIS COMMUNITY. VERIS – The Vocabulary for Event Recording and Incident Sharing. <http://veriscommunity.net/>, 2023. Last accessed 2023-05-31.
- [35] WAGNER, C., DULAUNOY, A., WAGENER, G., AND IKLODY, A. MISP: the design and implementation of a collaborative threat intelligence sharing platform. In *Proceedings of the 2016 ACM on Workshop on Information Sharing and Collaborative Security, WISCS 2016, Vienna, Austria, October 24 - 28, 2016* (2016), S. Katzenbeisser, E. R. Weippl, E. Blass, and F. Kerschbaum, Eds., ACM, pp. 49–56.
- [36] WEBSTER, J., AND WATSON, R. T. Analyzing the past to prepare for the future: Writing a literature review. *MIS Q.* 26, 2 (2002).
- [37] WEST-BROWN, M. J., STIKVOORT, D., KOSSAKOWSKI, K.-P., KILLCRECE, G., AND RUEFLE, R. Handbook for computer security incident response teams (CSIRTs). Tech. rep., Carnegie-Mellon Software Engineering Institute, 2003.
- [38] WROZEK, B., MAXIM, M., PROVOST, C., AND MCPHERSON, I. The external threat intelligence service providers landscape, Q1 2023. Tech. rep., Forrester Research, 2023.
- [39] YOUNG, A. L., AND YUNG, M. Cryptovirology: Extortion-based security threats and countermeasures. In *1996 IEEE Symposium on Security and Privacy, May 6-8, 1996, Oakland, CA, USA* (1996), IEEE Computer Society, pp. 129–140.
- [40] ZHAO, J., YAN, Q., LI, J., SHAO, M., HE, Z., AND LI, B. TIMiner: Automatically extracting and analyzing categorized cyber threat intelligence from social data. *Comput. Secur.* 95 (2020), 101867.

Part II

Research Papers

The second part of this dissertation contains publications P1 through P6, each prefaced by a short metadata summary.

1 A Comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective

Publication information

Current status: Published

Journal: IEEE Communications Surveys & Tutorials

Date of acceptance: 29 September 2021

Full citation: SCHLETTE, D., CASELLI, M., & PERNUL, G. (2021). A Comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective. *IEEE Communications Surveys & Tutorials*, 23(4), pp. 2525-2556.

Authors' contributions:	Daniel Schlette	70%
	Marco Caselli	20%
	Günther Pernul	10%

Journal description: IEEE Communications Surveys & Tutorials is an online journal published by the IEEE Communications Society covering all aspects of the communications field. It aims to be the premier source of peer-reviewed, comprehensive tutorials and surveys, and pointers to further sources.

A Comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective

Daniel Schlette¹, Marco Caselli², and Günther Pernul¹, *Member, IEEE*

Abstract—Cyber Threat Intelligence (CTI) is threat information intended for security purposes. However, use for incident response demands standardization. This study examines the broader security incident response perspective. Introducing 18 core concepts, we assist efforts to establish and assess current standardization approaches. We further provide the reader with a detailed analysis of 6 incident response formats. While we synthesize structural elements, we point to characteristics and show format deficiencies. Also, we describe how core concepts can be used to determine a suitable format for a given use case. Our surveys' findings indicate a consistent focus on incident response actions within all formats. Besides, playbooks are used to represent procedures. Different use cases suggest that organizations can leverage and combine multiple formats. Finally, we discuss open research challenges to fully realize incident response potentials.

Index Terms—Cyber threat intelligence, incident response, standardization, playbook format.

I. INTRODUCTION

THE COMPREHENSIVENESS of the Cyber Threat Intelligence (CTI) paradigm makes it ideal for coping with threats to information systems and information security. Commonly perceived as meaningful and actionable knowledge, CTI is based on structured, evidence-centered threat information [1], [2]. As such, threat intelligence is a central element to inform decision-makers about the current security status of their organization and to indicate necessary security measures.

Extensive research on CTI has defined its essential building blocks to comprise the threat information itself [3], [4], data formats [5], [6], [7], sharing and collaboration via dedicated platforms [8], [9], [10] as well as incident response [11], [12], all embraced by the topic of data quality [13], [14].

Starting with the underlying threat information, observable artifacts, Indicators of Compromise (IoCs) or Tactics, Techniques and Procedures (TTPs) form the content structured

Manuscript received February 1, 2021; revised July 9, 2021 and September 10, 2021; accepted September 29, 2021. Date of publication October 4, 2021; date of current version December 8, 2021. This work was supported in part by the Federal Ministry of Education and Research, Germany, as a part of the BMBF DEVISE Project, and in part by the Union's Horizon 2020 Research and Innovation Program under Grant 830927. (*Corresponding author: Daniel Schlette.*)

Daniel Schlette and Günther Pernul are with the Chair of Information Systems, University of Regensburg, 93053 Regensburg, Germany (e-mail: daniel.schlette@ur.de).

Marco Caselli is with the Department of Cybersecurity Technology, Siemens AG, 81739 Munich, Germany.

Digital Object Identifier 10.1109/COMST.2021.3117338

1553-877X © 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See <https://www.ieee.org/publications/rights/index.html> for more information.

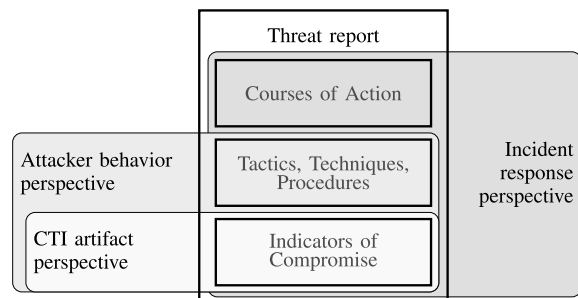


Fig. 1. Cyber Threat Intelligence Perspectives.

by CTI formats. Most notably, malware hashes and malicious IP addresses constitute CTI artifacts [15]. Indicated by recent studies, organizations might extract artifacts from unstructured data using mining techniques and analysis [16], [17], [18]. The representation enforced by CTI frameworks, standards, and other formats then supports various essential activities such as information sharing (and receiving) and incident response. As these are, in many ways, crucial domains for organizations, CTI sharing has been complemented with sharing platforms and concepts [19]. The incident response domain covers incident response processes and Courses of Action (CoAs) that constitute countermeasures to cyber attacks. Related incident reporting and early taxonomies [20] are also the historical roots of CTI. Lastly, the effectiveness of CTI for defensive purposes mandates data quality considerations due to the severe consequences of low-quality CTI. This multitude of facets makes up CTI and thus allows one to take on different perspectives on the paradigm (see Figure 1). Today, the most common CTI perspectives are on threat reporting, including informative description of CTI artifacts (e.g., IoCs) extended by attacker behavior (e.g., TTPs). In contrast, the perspective of incident response with its main advantage – to outline how to apply threat intelligence effectively – has not received a lot of research attention.

The situation is different when incident response is observed as a standalone concept. Most definitions of incident response approach the topic through its great practical relevance for organizations and its process focus [21], [22]. Encapsulated within incident response, information security incidents or imminent threats demand a reaction of some sort by the organization or individual under attack. This reaction is necessary to assure the functioning and security of its information systems. In this regard, ransomware that infected a customer database

2526

IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 23, NO. 4, FOURTH QUARTER 2021

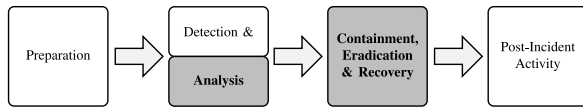


Fig. 2. Survey focus based on NIST Incident Response Life Cycle [23].

or a targeted intrusion on a critical manufacturing system endanger the business operations and can permanently threaten security. Adequate incident response will select and perform procedures to remove any malware, restore systems to a normal state and take precautions for future incidents. Blocking inbound network traffic or updating rules on attacker behavior in cyber defense systems are example procedures.

Typically, incident response describes a process with several phases. One of the most renowned frameworks – the incident response life cycle by the National Institute of Standards and Technology (NIST) [23] – starts with a *Preparation* phase, followed by *Detection & Analysis*, *Containment, Eradication & Recovery* and concludes with *Post-Incident Activity*. It is worth mentioning that between the four phases feedback loops exist. Other incident handling process models (e.g., CERT/CC [24], ITIL [25], [26], [27]) are in line with the NIST incident response life cycle. Nevertheless, often incident response is narrowed down to only the *Containment, Eradication & Recovery* activities, whereas incident management and incident handling provide the larger reference framework [21], [27]. We follow this more precise approach and center on the pivotal activities of incident response.

An elementary subarea in conjunction with incident response and its community is digital forensics. Digital forensics concerns data gathering and the detailed analysis of circumstances surrounding a security incident [26]. Within the NIST incident response life cycle, digital forensics mainly precedes the incident response action itself and can be attributed to *Detection & Analysis*. For our work, we separate between digital forensics and incident response and exclude the former. However, due to the nature of the analyzed data formats, there is at times overlap concerning investigative incident response activities. This situation leads to the focus of this survey described in Figure 2. The starting point of incident response and its standardization is hereby defined as trigger, alert, or event detected by an Intrusion Detection System (IDS), Security Information and Event Management (SIEM), or similar system, which then requires incident response actions. Also, CTI feeds, and structured threat reports are possible external starting points.

Beyond the structured process, incident response and its actions are built on additional cornerstones. People, processes, technology, governance, and compliance [28], [29] apply to incident response and manifest in its organizational integration. Organizations define Computer Emergency Response Teams (CERTs), Computer Security Incident Response Teams (CSIRTs) or Security Operations Centers (SOCs) to address operational security and incident response actions [30], [31]. Further, there is a data component relevant for incident response procedures which includes threat intelligence and other information from various sources [32]. As a result,

incident response links to CTI artifacts and is interwoven with the CTI paradigm.

The necessity of incident response standardization is emphasized by the recent U.S. *Executive Order 14028 - Improving the Nation's Cybersecurity* pointing to response playbooks [33]. Also, a major organizational security objective is swift reaction upon incident detection. Recent developments show that there is a community that pursues the move towards realizing this objective through incident response automation via software products and solutions [34]. Subsumed under the newly-coined term of *Security Orchestration, Automation and Response (SOAR)* a tremendous surge in vendors and products for CTI, SOCs and CERTs can be observed [35]. We derive that standardization and the inclusion of CTI artifacts are critical enablers of incident response automation. In addition, early work on incident response standardization and its connection with CTI demands further attention. It is the currently missing comprehensive coverage of countermeasure standardization in academic literature [36] paired with standardization developments that guided us towards this survey on incident response standardization.

This paper sheds light on existing standardization approaches for incident response and aims to pave the way for further advances beyond the status quo. The incident response perspective on CTI combines the inherent CTI focus on structured data formats and the domain of incident response with its active cyber defense. As the underlying standardization of incident response has remained largely uncovered, we contribute by identifying core concepts required for incident response. These core concepts can be categorized and emphasize essential characteristics mandatory for standardization approaches. Our contribution then extends to a comprehensive and detailed analysis of 6 incident response formats. Precisely, we analyze *Collaborative Automated Course of Action Operations (CACAO) for Cyber Security* [37], *Collaborative Open Playbook Standard (COPS)* [38], *Integrated Adaptive Cyber Defense (IACD) Framework* [39] as well as *Open Command and Control (OpenC2)* [40], *RE&CT Framework* [41], and *Resilient Event Conditions Action System against Threats (RECAST) Framework* [42]. Beyond the analyzed formats, we also document the larger product ecosystem.

Together with the description of the incident response formats, we outline how the core concepts are addressed and give a summary and recommendations for use. For further guidance, we contribute a side-by-side comparison of incident response formats and a format categorization. Any comparative analysis must take into account the way these formats will be used. For this purpose, our contribution to practical application is to indicate core concepts required for 3 separate use cases. More specifically, we show how the respective core concepts can be helpful to determine the best suitable incident response format for a given use case. The value of the incident response perspective and our survey is thus embedded largely in two parts – 1) theoretical basis (core concepts) and 2) analysis (format characteristics). These two parts lay the foundation for the many aspects of effective CTI use and incident response. The analysis of format characteristics reveals that playbooks and the structural concepts *Workflow, Actuator,*

TABLE I
ESSENTIAL CTI FORMATS

Category	Format	Inception	Maintainer	Alternative Formats
Frameworks	Lockhead Martin Cyber Kill Chain	2011	Lockhead Martin	MITRE ATT&CK
	MITRE ATT&CK	2013	MITRE	Cyber Kill Chains
Standards	Open Source Threat Intelligence Platform (MISP)	2011	EU & CIRCL	IODEF, VERIS, STIX
	Structured Threat Information eXpression (STIX)	2012	OASIS CTI TC	IODEF, VERIS, MISP
	Trusted Automated eXchange of Indicator Information (TAXII)	2012	OASIS CTI TC	Transportation methods
Scoring Systems	Common Vulnerability Scoring System (CVSS)	2005	FIRST	NCISS, CWSS
Enumerations	Common Platform Enumeration (CPE)	2007	NIST	SWID, PURL, SPDX
	Common Vulnerabilities and Exposures (CVE)	1999	MITRE	OVAL
	Common Weakness Enumeration (CWE)	2008	MITRE	CAPEC

Action, and *Artifact* are essential to organize incident response, but their implementation varies.

The outline of this survey is as follows. In the next section, we introduce essential data formats found in CTI and present relevant background information, related work on CTI format analysis and incident response leading to incident response formats and the surrounding product ecosystem. In Section III we derive and describe in-depth incident response core concepts necessary for incident response format analysis categorized as either 1) general, 2) structural, 3) technological, or 4) security concepts. Detailed description and analysis of incident response formats based upon the identified core concepts constitute Section IV. Relevant findings highlighting various deficiencies and gaps in the incident response formats are thereupon discussed in Section V. As the incident response formats will eventually serve a particular use case, we discuss in Section VI core concepts relevant for the use cases of automating, sharing, and reporting incident response capabilities. Section VII concludes the paper.

II. CYBER THREAT INTELLIGENCE FORMATS

In this section, we introduce prevalent CTI formats. We briefly discuss terminology and different CTI formats categorized according to characteristics of use. Related work provides means of format analysis and indicates a research gap with regard to incident response standardization. We, therefore, emphasize incident response formats and related approaches.

A. Categorization

The term *data format* is used throughout this paper to refer to logically and semantically structured data and information. We acknowledge the differences between types of structured information such as frameworks and serialization schemes as granularity and technicality vary. The categorization approach of CTI formats seen in Figure 3 highlights usage and includes the high-level framework category aimed to fulfill security guidance requirements. Next, dedicated CTI standards align on a spectrum between representational and operational use. While most CTI standards are ratified by standardization bodies, the standards category also centers on the criteria of comprehensiveness and data structuring. The more granular data formats categorized as scoring systems and security enumerations contain fewer or more condensed information and

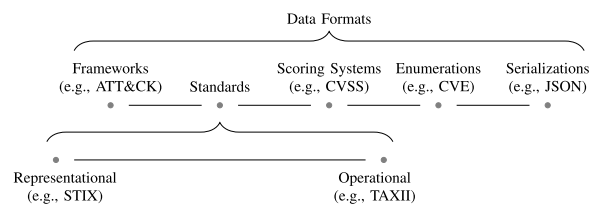


Fig. 3. Categorization of Cyber Threat Intelligence Data Formats.

a simpler structure. With serialization schemes, the technical basis of many higher-level formats is also part of the categorization. It is worth noting that the categorization derived from existing CTI formats, specification documents, and few related approaches [5], [12], [43] might not apply to other domains.

Based on the extensive research and development conducted on CTI formats, the following categorization includes an overview of the most essential CTI formats. Additionally, basic details of these formats are briefly summarized in Table I.

1) *Frameworks*: The objective of CTI frameworks is to provide an overview of specific threat characteristics. Most frameworks include elements for chronological structuring and are broad in scope. Organizations can extract relevant knowledge from frameworks according to individual needs.

Two prominent frameworks in the field of CTI are the Lockheed Martin *Cyber Kill Chain* [44] and the *MITRE ATT&CK* framework [45]. Both aim to describe adversary behavior in the various stages of an attack. From a cyber defense perspective, frameworks can be leveraged to identify gaps in an organization’s security posture and to build relevant knowledge. TTPs represent one possible structuring level of these data formats.

2) *Standards*: The objective of CTI standards is to provide a comprehensive methodology to describe threats, attacks, and security incidents in all their facets. Nevertheless, CTI standards can have specific focal points. Besides the representation of security information, CTI standards can also be intended for specific operational use cases.

Among the comprehensive and ratified CTI standards is the *Open Source Threat Intelligence Platform (MISP)* format [8]. The MISP core format follows a flexible approach to CTI description based on event, attribute and tag objects [46]. *Structured Threat Information eXpression (STIX)* is another established and widely used graph-based CTI standard [47].

In its newest version, STIX2.1, the format specifies multiple STIX Domain Objects (SDOs) and STIX Cyber-observable Objects (SCOs) available for connected CTI representation [48]. Whereas STIX2.1 envisions coverage of incident response elements in the form of *Course of Action (CoA)* objects, these remain unspecified. For operational use, STIX is accompanied by the *Trusted Automated eXchange of Indicator Information (TAXII)* format [49]. TAXII supports CTI sharing with its client-server model [50].

3) *Scoring Systems*: The objective of CTI scoring systems is to provide an indicative metric for security implications of the artifact under assessment. Scoring systems typically include a formal component enabling the calculation of the respective score. This precise quantitative expression can then be used for organizational decision-making.

Scores adhering to the *Common Vulnerability Scoring System (CVSS)* range from value 0 to 10 and contain relevant information about the characteristics and significance of a given vulnerability [51].

4) *Enumerations*: The objective of security enumerations is to provide unique identifiers (IDs) to specific security artifacts. Most security enumerations are based on a clearly defined and simplistic representation. A unique ID is hereby composed of or supplemented by essential artifact characteristics.

For classes of IT assets, unique representation is often based on the *Common Platform Enumeration (CPE)* [52]. Further, vulnerabilities are addressed by the *Common Vulnerabilities and Exposures (CVE)* enumeration [53]. A third essential enumeration, the *Common Weakness Enumeration (CWE)*, is focused on software flaws [54].

5) *Serializations*: The objective of serializations is to provide schemes for transferring and storing data in a byte stream. In CTI, *JavaScript Object Notation (JSON)* and *eXtensible Markup Language (XML)* are widely used serializations.

B. Related Work

Threat intelligence formats have been thoroughly analyzed and covered in multiple research publications as interest from practitioners and researchers increased significantly in recent years. Besides, several surveys emphasize the importance of the underlying data formats used for representation and CTI sharing. We, therefore, group relevant research into two groups: 1) CTI format analyses and 2) surveys. The former group covers related work on CTI formats with comparative elements and in-depth format considerations. The latter group provides the necessary positioning of CTI formats in the wider context of CTI and incident response.

In chronological order of publication, CTI format analyses include the early work by Fenz *et al.* [55] evaluating the semantic potential of CTI formats, for instance, the Incident Object Description Exchange Format (IODEF). As another starting point, Hernandez-Ardieta *et al.* [56] aggregate additional CTI formats derived from the *Making Security Measurable* MITRE project. Dandurand *et al.* [57] from the European Union Agency for Cybersecurity (ENISA) shed light on the topic with an extensive yet not deep examination of a multitude of CTI formats. Analysis and evaluation of

CTI formats are further pursued by Steinberger *et al.* [6]. Here, for the first time, numerous detailed evaluation criteria are specified and applied to CTI formats. Based on a model describing the various elements of CTI, Mavroeidis and Bromander [12] conduct a detailed structural evaluation of CTI formats. The components for structural evaluation include attack countermeasures intended for incident response and represented by a CoA element. As the evaluation reveals, only a few CTI formats (e.g., STIX) even consider incident response. A detailed comparative analysis of more recent CTI formats by Menges and Pernul [7] combines and extends existing evaluation criteria. The authors enhance previous CTI comparisons by emphasizing strengths, weaknesses, and structure as well as use cases for CTI formats [58]. Other current works reproduce CTI format analysis with similar evaluation criteria and results [59], [60] or extend research to the evaluation of CTI sharing platforms. In this respect, Bauer *et al.* [61] identified the necessity of CTI standardization for information description and CTI sharing use cases within their non-functional platform criteria.

Influential surveys on CTI and incident response include, above all, research of Skopik *et al.* [11] on the CTI ecosystem at large. Thereby data formats and standardization form one dimension as the authors focus on a comprehensive set of dimensions of security information sharing and incident response. In the same direction, the survey of Wagner *et al.* [62] aggregates existing knowledge on CTI sharing. While outlining sharing elements, data formats are considered beneficial for efficient knowledge dissemination and incident response. Ab Rahman and Choo [21] investigate different incident response process models. Their work provides a basis for understanding incident response and also contains response strategy types. Finally, Nespoli *et al.* [36] derive a framework for optimal countermeasures selection against cyber attacks. The framework includes atomic countermeasure options and actions for which the authors identify a lack of standard representation. It can be observed that established CTI models and data formats partially foresee incident response. However, in contrast to this paper, no comprehensive analysis of incident response standardization has been conducted. Therefore we build on related work of existing and well-researched CTI formats to analyze incident response formats in-depth.

C. Incident Response Formats

Incident response formats exist but have yet to evolve and receive further attention. Whereas other formats have gradually become part of comprehensive CTI standards, the few incident response formats remained separate. However, recent developments concerning incident response formats and related *Security Orchestration, Automation and Response (SOAR)* products indicate growing maturity.

In the following, we focus on specific incident response data formats. These formats are part of a larger surrounding ecosystem of incident response displayed in Table II. For completeness, we also list and briefly describe general utility data formats, digital forensics formats, and SOAR products (see

TABLE II
INCIDENT RESPONSE FORMATS AND PRODUCTS

Category	Format / Name	Source	Inception	Maintainer / Vendor	Serialization	License	Analysis
General Utility	Ansible	[63]	2012	Red Hat	YAML	GPLv3.0	×
	BPMN2.0	[64]	2001	OMG	XML	OMG License	×
	OpenDXL	[65]	2016	McAfee	JSON	Apache 2.0	×
	ROLIE	[66]	2012	IETF	XML	IETF License	×
Digital Forensics	AFF4	[67]	2009	Individual	Turtle	GPLv1.3	×
	DFXML	[68]	2012	NIST	XML	CC0 1.0 / LGPL	×
Incident Response	CACAO	[37]	2017	OASIS	JSON	OASIS Open	✓
	COPS	[38]	2016	DEMISTO	YAML	MIT	✓
	IACD	[39]	2014	DHS / NSA / JHU	XML	CC BY 4.0	✓
	OPENC2	[40]	2015	OASIS	JSON	OASIS Open	✓
	RE&CT	[41]	2019	ATC Project	YAML	Apache 2.0	✓
	RECAST	[42]	2018	MITRE	N/A	N/A	✓
SOAR Product	ArcSight SOAR	[69]	2017	Micro Focus	N/A	Proprietary	×
	Ayehu NG	[70]	2007	Ayehu	N/A	Proprietary	×
	Cortex XSOAR	[71]	2015	Palo Alto Networks	N/A	Proprietary	×
	D3 SOAR	[72]	2004	D3 Security	N/A	Proprietary	×
	Dragos Platform	[73]	2016	Dragos	N/A	Proprietary	×
	EclecticIQ	[74]	2014	EclecticIQ	N/A	Proprietary	×
	FortiSOAR	[75]	2011	Fortinet	N/A	Proprietary	×
	Helix	[76]	2017	FireEye	N/A	Proprietary	×
	IncMan SOAR	[77]	2013	DF Labs	N/A	Proprietary	×
	InsightConnect	[78]	2017	Rapid7	N/A	Proprietary	×
	ONAP	[79]	2017	The Linux Foundation	N/A	Apache 2.0	×
	Playbook Viewer	[80]	2017	Unit 42	JSON	MIT	×
	Resilient	[81]	2010	IBM Security	N/A	Proprietary	×
	Resolve	[82]	2014	Resolve	N/A	Proprietary	×
	Security Operations	[83]	2014	ServiceNow	N/A	Proprietary	×
	Shuffle	[84]	2019	Individual	N/A	MIT & AGPLv3.0	×
	Siemplify	[85]	2015	Siemplify	N/A	Proprietary	×
	SOAR+	[86]	2016	LogicHub	N/A	Proprietary	×
	SOCAutomation	[87]	2005	Honeycomb	N/A	Proprietary	×
	Splunk Phantom	[88]	2014	Splunk	N/A	Proprietary	×
Swimlane SOAR	[89]	2014	Swimlane	N/A	Proprietary	×	
TheHive & Cortex	[90]	2014	TheHive Project	JSON	AGPLv3.0	×	
ThreatConnect SOAR	[91]	2011	ThreatConnect	N/A	Proprietary	×	
ThreatStream	[92]	2013	Anomali	N/A	Proprietary	×	
ThreatQ	[93]	2013	ThreatQuotient	N/A	Proprietary	×	
Tines	[94]	2018	Tines	N/A	Proprietary	×	
Virtual Cyber Fusion	[95]	2018	Cyware	N/A	Proprietary	×	
WALKOFF	[96]	2016	NSA Cybersecurity	JSON	CC0 1.0	×	

Section IV-G). However, we refrain from detailed analysis due to data availability (SOAR products), focus (digital forensics), and expediency (general utility). For instance, SOAR products include proprietary characteristics which hinder assessment. Digital forensic formats are related but not at the center of incident response. Thus, despite their partial relevance, we provide detailed analyses for six incident response formats only.

Following the inception of the *Integrated Adaptive Cyber Defense (IACD) Framework* [39] in 2014, subsequently, the formats *Open Command and Control (OpenC2)* [40], *Collaborative Open Playbook Standard (COPS)* [38], *Collaborative Automated Course of Action Operations (CACAO) for Cyber Security* [37], *Resilient Event Conditions Action System against Threats (RECAST) Framework* [42] and *RE&CT Framework* [41] have been introduced (see Figure 4).

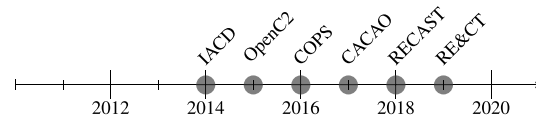


Fig. 4. Timeline of Incident Response Formats (first mention).

III. INCIDENT RESPONSE CORE CONCEPTS

Based on our initial analysis of incident response, we identified relevant concepts. These core incident response concepts allow for classification and comparison of the individual formats and are first briefly introduced. In Table III we list concept categories, core concepts, and derived capabilities that are supported by the respective concept. Derived capabilities

TABLE III
INCIDENT RESPONSE CONCEPTS AND DERIVED CAPABILITIES

Category	Core Concept	Derived Capabilities
General	Aggregability	Information Sharing, Semantics
	Categorization	Comprehensibility, Delimitation
	Granularity	Structuring
	Versioning	Data Quality, Maintenance
	Referencing	Usability, Separation
	Extensibility	Customization, Sustainability
	Readability	Comprehensibility, Interpretability
Structural	Unambiguous Semantics	Clarity, Interorganizational Understanding and Application
	Workflow	Sequencing, Operations
	Actuator	Actionability
	Action	Atomicity
Technological	Artifact	CTI Integration
	Community	Usability, Acceptance, Maintenance
	Application	Technical Integration, Interoperability
Security	Serialization	Data Storage, Data Transfer
	Confidentiality	Information Sharing, Operations
	Authorization	Misuse Prevention, Operations
	Prioritization	Information Importance, Operations

are intended to illustrate additional user requirements associated with the core concepts. For the core concepts, previous analyses of data formats in CTI arrived at slightly different comparison criteria [6], [7], [55]. We put stronger emphasis on conceptual elements with our approach while still subsuming existing criteria within defined core concepts. Wherever possible, we incorporated definitions and naming conventions of established concepts. However, aggregation of concepts and incident response specifics demands new concepts and new concept names. We chose core concepts to represent distinct areas of incident response, yet at times, core concepts can overlap.

In the following, the categorized core concepts are described in detail. We first provide a brief description of each concept in Table IV and highlight examples of implementation in incident response formats. Besides, we indicate whether or not a concept is present in encompassing CTI. As incident response is part of CTI, a multitude of concepts is inherited. With regard to specific structural concepts, the ones found in incident response differ primarily in the level of detail compared to CTI. These structural concepts, as well as the concept of authorization, are marked accordingly. Hereinafter, we focus on a deeper understanding of each concept before we later analyze incident response formats.

A. General Concepts

We identified a group of general concepts related to incident response standardization. These general concepts consider incident response information itself and the structured

TABLE IV
INCIDENT RESPONSE CORE CONCEPTS DESCRIPTION

Core Concept	Description	Example(s)	CTI
Aggregability	Grouping of related incident response elements	playbook	✓
Categorization	Distinguishable objectives of incident response	stage, playbook type	✓
Granularity	Different levels of incident response information	workflow, workflow step, command, action	✓
Versioning	Documenting incident response information updates or revocations	metadata, change mechanism	✓
Referencing	Referral to incident response elements with (unique) IDs	uuid, enumeration	✓
Extensibility	Provision of additional incident response information	open vocabulary, external source	✓
Readability	Legibility of incident response information	human, machine	✓
Unambiguous Semantics	Distinct meaning of different incident response elements	component definition, instantiation	✓
Workflow	Procedural ordering of incident response actions	instruction list, process	×
Actuator	Subject executing an incident response action	system, human expert, field	×
Action	Executable element of incident response	item, command	×
Artifact	Object of incident response action	variable, target, CTI element	×
Community	Supporting elements of incident response standardization	GitHub repository, documentation, collaboration	✓
Application	Technological dependencies of incident response standardization	proxy layer, direct conversion	✓
Serialization	Encoding of incident response information	JSON, XML, YAML	✓
Confidentiality	Sensitivity aspects of incident response information	data marking, privacy	✓
Authorization	Control measures of incident response procedures	ownership, sandboxing, impact	×
Prioritization	Urgency expression of incident response actions	scoring, severity	✓

✓ CTI origin × not in CTI

representation of this information in incident response formats. We mention the typical representing artifact in incident response for each general concept (e.g., playbooks enabling aggregability).

1) *Aggregability (Playbook)*: A key concept of incident response standardization is the ability to group or bundle elements on various levels. Aggregability, in general, implies different forms of semantic or logical aggregation and supports information sharing. Inspired by traditional CTI and

threat reports, playbooks represent the concept of aggregability within incident response [97]. These high-level constructs allow their creators to bundle incident response information subjectively. Parallels of playbooks are not only found in the STIX2.1 CTI format (i.e., `report` object) but also reflect software development (e.g., libraries, modules, and classes). As incident response standardization aims to capture previously unspecified incident response concepts, it is reasonable to include playbooks in designated data formats. Playbooks allow to define, reuse and archive incident response processes and information adapted to a specific context. The characteristic of playbooks within incident response to also contain structural elements that impose the ordering of actions is later covered in the workflow concept (see Section III-B9) [98].

2) *Categorization (Objective)*: Incident response tasks can fulfill different objectives. There are four overarching categories – *investigation, mitigation, remediation, and prevention* – which represent aims of incident response actions derived from incident handling recommendations [23]. Thus, categorization builds a core concept of incident response standardization as it supports comprehensibility via more precise definitions and delimitation of actions. The naming of the categories intuitively indicates the following definitions.

- Investigation – Actions that gather essential information and mainly answer the questions “What has happened to an IT-System?” and “How has it happened?”
- Mitigation – Actions that respond to information security incidents or other existing problems and reduce the negative impact and follow-up problems of such events.
- Remediation – Actions that ultimately fix a problem or eradicate existing flaws and return impacted systems to a clean state.
- Prevention – Actions that help to avoid unwanted events to occur and serve as defensive measures.

The definitions of these objectives, however, are not without overlap and should only provide some guidance. Formats may choose a different categorization or introduce categories before or more granular than those described above (e.g., detection or lessons learned). The detection of security incidents, in particular, is a task regularly conducted by SOC personnel and thus arguably not genuine to incident response standardization and its formats.

3) *Granularity (Technical and Non-Technical Information)*: Incident response standardization bridges the gap between CTI and its use for countermeasures. CTI features different levels of information. It describes both low-level observable objects (e.g., hash values, IP addresses) and other IoCs as well as attribution elements and attack patterns. Incident response standardization likewise makes use of the granularity concept to structure information. Here, the information levels allow top-down or bottom-up approaches based on overall directives for incident response processes or use of technical CTI in specific commands and actions. As a consequence, different recipients can receive incident response information configured to their needs.

4) *Versioning (Metadata)*: Similar to comprehensive CTI standards, processes and changes to information play a role

in incident response standardization and support data quality. Incident response information is generated, applied, modified, and eventually revoked. Revocation constitutes a crucial component of the incident response information life cycle as it implies a final and definite form of representing information. Versioning considers the different possible life cycle stages and is integrated into incident response formats via metadata. As a result, attributes capture information life cycle stages through the use of timestamps. Mechanisms to test, modify and maintain information (e.g., merging data from different sources) embed the continuous vetting process of relevant information into incident response formats. The rules on how to proceed with versioning depend on the criteria specified by each format. The main aspect is how to cope with extensions by either generating a new object or modifying an existing one.

5) *Referencing (Identifier)*: Referencing builds another crucial concept of incident response standardization. First of all, it supports usability and the separation of procedural and technical elements. In CTI standards (e.g., STIX2.1), the concept of referencing is implemented with unique identifiers, enumerations (i.e., naming conventions), or open vocabularies (i.e., lists of values for specific properties). Incident response formats also employ these concepts. Internal referencing in incident response standardization allows reusing processes, objects, and standardized representations based on their identifiers or naming convention. However, also external object referencing can be found. External referencing integrates incident response standardization in the comprehensive CTI environment and supports leveraging existing information and standards without reimplementing or redefining them. Thus, a layered approach with self-contained, reusable elements on lower levels becomes possible.

6) *Extensibility (Open Vocabulary)*: The concept of extensibility goes beyond referencing and introduces mechanisms for new context-based or user-based definitions and objects. Extensibility supports the customization of an incident response format and its sustainability as requirements change over time. In this regard, changes in the encompassing CTI ecosystem might trigger necessary adaptations. Open vocabularies are a common implementation of the extensibility concept as users can provide additional information for elements of their choice. Besides open vocabularies, external sources are integrated into incident response formats. For example, a Uniform Resource Locator (URL) can point to relevant external information on the Internet. Due to the broader scope of CTI, its formats also allow the introduction of entirely new structural elements (e.g., STIX Domain Objects). Incident response formats might incorporate similar mechanisms.

7) *Readability (Human/Machine)*: Readability is targeted at humans or machines and constitutes an essential concept of incident response standardization. Automation aspects of incident response directly pertain to machine readability, whereas organizational aspects put focus on readability by human decision-makers. Therefore, either humans or machines must be able to read incident response information structured according to a given incident response format. The concept of readability supports the comprehensibility and interpretability of incident response information. To conduct incident response

2532

IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 23, NO. 4, FOURTH QUARTER 2021

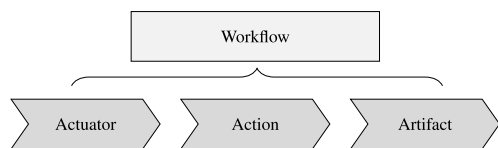


Fig. 5. Structural Incident Response Concepts.

effectively, readability is an integral part. In incident response formats it is largely influenced by serialization schemes and other forms of representation such as markup languages.

8) *Unambiguous Semantics (Definition)*: Data formats provide a structured framework to express semantics. The concept of unambiguous semantics comprises elements of incident response standardization that foster clarity and avoid ambiguities. While difficult to assess, unambiguous semantics support the inter-organizational understanding and application of the information contained in an incident response format. Ambiguities in incident response formats concern structural concepts and object definitions. For instance, the *target* object found in different formats has various meanings and thus demands a semantic analysis and definition.

B. Structural Concepts

Incident response standardization is founded on structural concepts. Figure 5 depicts four identified structural concepts and their logical relations. In essence, a workflow is used to contain actuators, actions, and artifacts of incident response.

9) *Workflow*: The term incident response implies that there is an reaction of some sort to an event. This reaction is represented and organized by the workflow concept and in most cases based on three structural elements of incident response. On an abstract level each workflow consist of a subject, a verb and an object. In the context of incident response this 3-tuple can be specified as an actuator performing an action on a given artifact. A workflow then captures multiple sequential or parallel aligned 3-tuples that form the incident response procedure. Within a workflow, individual elements are ordered and aligned based on either logical or temporal conditions. In contrast to the procedural life cycle information included in the versioning concept, the workflow concept addresses sequencing of multiple actions and supports operations. Workflows therefore share characteristics with algorithms and instruction sets. Incident response formats implement the workflow concept differently, introduce their own naming conventions and combine or omit some of its elements. In general, the more incident response formats focus on precise actions the less attention is paid to the workflow concept.

10) *Actuator*: For each incident response there is an entity that executes the process step, which we refer to as *Actuator*. Incident response information is always directed at a specific actuator to act upon the information. If there is no actuator, countermeasures to security incidents and attacks cannot be effectively processed and executed. Hence, the actuator constitutes another essential concept of incident response and supports actionability of incident response information contained within CTI. Information systems are common actuators

and incident response standardization is closely related to the use of defensive technologies and tools. Nevertheless, incident response standardization adheres to the well-known information security paradigm also incorporating people and processes which are manual actuators. For instance, responsibilities and organizational attack countermeasures are best performed by security experts or certain roles.

11) *Action*: Actions define precise incident response measures and are technical or non-technical depending on the associated actuator. The concept of actions in incident response standardization aims to achieve atomicity. Therefore, actions have a clear scope. Additionally, incident response formats relate relevant execution information with the action concept. Here, timing arguments and executable commands are prominent examples.

12) *Artifact*: Artifacts represent the objects of incident response actions. The structural artifact concept fosters the integration of CTI in incident response standardization. In particular low-level observables (e.g., domain names or IP addresses) serve as artifacts. However, not all incident response formats separate actuators, actions and artifacts. It can thus be observed that some of the structural concepts (e.g., action and artifact) are indistinguishably merged together.

C. Technological Concepts

Technological concepts foster the maturity of incident response and help format use. Similar to CTI, we identify the concept of community with elements supporting incident response standardization in general and a given format in particular as relevant. A stronger focus on applying incident response information compared to describing and sharing CTI leads us to introduce a technical oriented concept of application. Finally, serialization is omnipresent when analyzing data formats and is thus included for incident response formats.

13) *Community*: The community concept is a necessary element of incident response formats to reach acceptance and widespread use. Supporting aspects of incident response standardization and its technological foundations are therein comprised. The community concepts covers the mutual development and collaboration on incident response formats and supports usability. Detailed documentation, best practices and openly accessible knowledge repositories are cornerstones of any practical application of incident response formats and technologies. With licensing terms and maintenance efforts the community concept further addresses legal concerns and continuous suitability of implementation.

14) *Application*: Incident response and its standardization concern the application of relevant incident response information. The act of using incident response information involves applications, tools and systems already in use. Based on the concepts of actuator (Section III-B10), action (Section III-B11) and artifact (Section III-B12) the application concept in incident response supports technical integration, interoperability and addresses external dependencies. Depending on the structuring data format and accompanying mechanisms, incident response application is performed directly or indirectly. Direct use of incident

response information demands a direct conversion of a given, technology agnostic, data format to actuator or device specific protocols and connectors. As an example, incident response formats and frameworks may already provide their data in multiple vendor specific formats and thus incorporate external dependencies to SIEM systems. The indirect approach makes use of a proxy layer handling integration with technologies and tools. This proxy layer receives incident response data and then performs appropriate conversion and transfer to actuators.

15) *Serialization*: Serialization incorporates elements of data encoding in incident response standardization. This is necessary to support data storage as well as exchange and transfer of information via networks. Whereas serialization is often-times a mandatory part of incident response format implementation, the specification of the formats is independent of serialization. Human-readability and machine-readability are two aspects in close relation with the chosen serialization schema as serialization influences legibility. Incident response formats mostly use JavaScript Object Notation (JSON)¹ and YAML Ain't Markup Language (YAML)² serialization schemes.

D. Security Concepts

Security concepts further define incident response. Here, security concepts target the incident response information being presented. We identify confidentiality as an important security concept due to implications resulting from access to incident response information. It is worth mentioning that beyond formats, the topic of privacy is crucial for incident response. However, as privacy is a highly organization-specific and use case-centric topic it is not directly present in incident response formats. Therefore, confidentiality captures any generic privacy aspects. Additionally, incident response information is about organizations using it. The concepts of authorization and prioritization are thus two relevant security concepts.

16) *Confidentiality*: Incident response information is often sensitive as it pertains to countermeasure specifics, processes and security incidents. Sharing and using this information internally or externally demands measures captured by the confidentiality concept. Confidential incident response information must be clearly marked and handled appropriately. Without adequate confidentiality inter-organizational use of incident response formats is not warranted. Therefore, the confidentiality concept supports operations and the acceptance of incident response standardization in the first place. Confidentiality measures included in incident response formats are data markings that allow to define levels of confidentiality. A common example is the use of the Traffic Light Protocol (TLP) indication.

17) *Authorization*: Incident response standardization use cases (e.g., automation) can have security implications. The concept of authorization describes approval mechanisms in incident response formats. Various authorization measures support the prevention of intentional or unintentional misuse of incident response information. For instance, it is advisable

¹<https://www.json.org>

²<https://yaml.org/spec/1.2/spec.html>

TABLE V
INCIDENT RESPONSE FORMAT ANALYSIS APPROACH

Category	Description	Level
Name	Descriptive term	
Abbreviation	Descriptive, short identifier	
Main objective	Distilled overall objective	
Inception	Year of first publication	Basics
Maintainer	Organization in charge of development	
Standardization	Standardization body (aimed for)	
License	Intellectual property rights	
Serialization	Technical implementation procedure	
Objective details	1-3 objective descriptions	Aims
Academic literature	Research papers & books	Stats
Gray literature	Additional documents & white papers	
Latest developments	Meetings, publications & visibility	

for organizations to document the potential impact of incident response procedures. In addition, assigning responsibilities and considering further pitfalls of incident response actions can help to limit the attack surface. Hence, incident response formats can provide specific properties and integrate information for authorization.

18) *Prioritization*: Not all incident response information must be treated equal. As there are severe and less severe security incidents the prioritization concept is relevant for incident response formats [99]. In general, prioritization expresses the urgency of incident response execution relative to other incident response procedures. Prioritization supports the information importance and operations related to incident response. Within incident response formats, prioritization is mostly realized with indicating severity.

IV. INCIDENT RESPONSE FORMAT ANALYSIS

Our approach to analysis of incident response formats is split in two parts. First, we provide a detailed and systematic overview of each analyzed format according to the characteristics in Table V. This overview contains basic information about the incident response format, information about its aims and a rough statistical estimate of publications as well as latest developments.

The second part is centered on a thorough analysis of each format according to the core incident response concepts established earlier (see Section III). This conceptual analysis is intended to highlight specifics of each incident response format and serves as a basis for comparison. We conducted the analysis in late 2020 and early 2021 reflecting the current state of incident response formats at that time.

A. Collaborative Automated Course of Action Operations (CACAO) for Cyber Security

Version: CACAO Security Playbooks Version 1.0 – Committee Specification 01 [37].

Basics: Generic incident response automation via structured playbooks is the objective of the *Collaborative Automated Course of Action Operations (CACAO) for Cyber Security* data format. First initiated as Internet Engineering Task Force (IETF) draft in 2017 CACAO is currently maintained by the nonprofit Organization for the Advancement

2534

IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 23, NO. 4, FOURTH QUARTER 2021

of Structured Information Standards (OASIS). A dedicated technical committee pursues and oversees the development towards an original standard under permissive Intellectual Property Rights (IPR) policy by OASIS. Eventually, ratification will include OASIS and other potential standardization bodies while CACAO envisions a JSON serialization.

Aims: CACAO describes a first attempt to advance and standardize actions taken in the context of threat intelligence and incident response. While still in early stages CACAO must be seen as a proposal towards a more precise but necessary structured definition of countermeasures. Further objectives of CACAO are automation and cross-technology as well as interorganizational operation. This includes to formalize both data format and data sharing of the CoA concept immanent to CTI. Special focus of CACAO is on security playbooks containing procedural logic and multiple actions.

Statistics: As of January 2021 CACAO matured from draft status to the current specification which serves as point of reference for the format [37]. Information about CACAO can refer to few additional sources.

- Academic literature on CACAO is almost not existent. For CACAO and the following analyzed formats we conducted a key word search in common academic literature databases (e.g., ACM Digital Library, IEEE Xplore, SpringerLink, DBLP, etc.) including forward and backward search. A single paper published in the proceedings of the International Telecommunication Union (ITU) Kaleidoscope conference very briefly describes CACAO and its envisioned position within CTI automation [100]. For completeness, a newly published book on Internet standards covers OASIS and thereby lists among its many other standards CACAO [101].
- Gray literature on CACAO includes the original IETF Internet-Draft charter and introduction.³ Besides, there is an OASIS working document, the approved *Security Playbook Requirements* [102], outlining standard requirements.
- Latest developments around CACAO included the progress towards the completion of the working draft. The current state of CACAO can be retrieved from the technical committee.⁴ The ratification by this OASIS committee and publication of the specification achieved in January 2021 constitute an important milestone.

General Concepts: The CACAO format covers previously introduced core concepts of incident response standardization to varying extent. Above all, playbooks, workflow steps, commands, targets, extensions and data markings represent object classes in CACAO to realize automated incident response. These structural elements are complemented by supporting concepts necessary for adequate standardization.

The *Aggregability* concept in CACAO is based on playbooks. These playbooks either contain precise and ready-to-use information or represent template documents to inform about exemplary actions related to security incidents. The

CACAO specification mentions events that trigger playbooks but does not implement a specific property.

Within CACAO *Categorization* is defined as part of its terminology and also included as playbook type enumeration. Playbook objects must implement a type attribute using a predefined value of the enumeration. In its approach to categorization CACAO specifies a detection playbook for orchestrating detection without elaborating on a specific detection action.

On a structural level *Granularity* manifests in CACAO workflows, workflow steps and commands. Whereas workflows and workflow steps capture procedural logic and center on organizational processes, commands represent more technical information. Besides, the CACAO format allows detailed expression of incident response elements through many optional attributes. It is possible to express rather manual, investigative and informative tasks in CACAO as well as precisely executable information.

CACAO playbooks contain metadata. Timestamps document creation and modification of elements and embed the *Versioning* concept in CACAO. To revoke information an additional attribute can be used. CACAO follows other data formats providing guidance on object creation and republication thus limiting misinterpretation. An early architecture model of CACAO further outlines lifecycle aspects of verification, monitoring and reporting. However, it is left to applications using the CACAO format to deal with versioning ambiguities, outdated information and other data quality problems.

For internal *Referencing* CACAO uses Universally Unique Identifiers in version 5 (UUIDv5) as defined by Request For Comments (RFC) 4122. Each CACAO object is identifiable by `object-type-UUID`. In addition, referencing is integrated in CACAO objects. Playbooks refer to workflow steps, targets, extensions and data markings. Workflow steps refer to following workflows steps, commands, targets or other playbooks. And on the lowest level commands and targets refer to variables (e.g., IP address) inherent in a given playbook. CACAO also assists referral to enumerations with predefined attribute values.

The concept of *Extensibility* in CACAO centers on open vocabularies and external information. Open vocabularies allow users to introduce definitions for attribute values. For example, the `command-type-ov` and the `target-type-ov` capture command and target types. External sources are also supported to some extent. STIX2.1 identity objects document playbook creators and extension objects can enhance CACAO objects with complementary information.

The *Readability* concept is highly subjective when considering human-readability. Incident response information expressed with the CACAO format is presented in JSON. As JSON intends to foster machine processing this decision documents the automation focus and machine-readability. In contrast, human-readability is given by the specification document but not the data format itself. Thus, a thorough understanding of CACAO-described information requires human analysts to be supported by dedicated tools especially when coping with larger JSON documents.

³<https://datatracker.ietf.org/wg/cacao/about/>

⁴https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=cacao

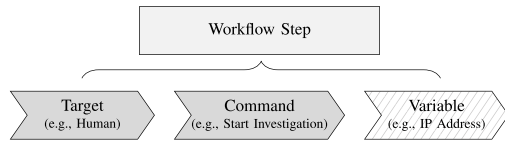


Fig. 6. Structural Description of CACAO Playbooks.

Negative effects when using the CACAO format caused by ambiguity are reduced with a terminology section and object class definitions. The CACAO specification addresses *Unambiguous Semantics* by defining six object classes and associated mandatory and optional attributes. CACAO naming conventions not always intuitively align with the objects' semantics. Also, the definition of CACAO objects is in parts still vague and leaves some room for interpretation (e.g., commands). However, over the course of standard development concepts (e.g., action object) have been eliminated to avoid redundancies.

Structural Concepts: The analysis of the aforementioned generic structural concepts – *Workflow*, *Actuator*, *Action* and *Artifact* – is implemented by CACAO adhering to a different naming convention. In the following, CACAO's *Workflow Step*, *Target* and *Command* object definitions as well as the *Variable* concept depicted in Figure 6 are analyzed. An exemplary CACAO workflow step might consist of a human starting investigation of an IP address. It is worth noting, that variables are part of CACAO but do not represent a clearly defined object class or artifact concept. We therefore opted to illustrate the lack of definition using gray lines within its structural description (Figure 6). The same applies to other incident response formats if structural concepts are incomplete.

In CACAO workflow steps represent the *Workflow* concept. Different types of workflow steps (e.g., start, if-condition, parallel, etc.) introduce temporal and conditional logic through specific attributes. For the most granular step – single action step – attributes capture targets and commands for execution. To realize batch processing multiple targets and multiple commands can be defined. All workflow steps support timeouts or delays as well as feedback mechanisms with information on how to proceed in case of success or failure.

Target objects of CACAO cover the *Actuator* concept. A target is defined as entity, system or device to handle incident response information in form of commands. CACAO specifies target types and thereby reaches from organizational entities (individual, group, organization) to geographical entities (location or sector) and to security infrastructure as well as network elements. Depending on the target type, specific attributes foster correct execution and identification. For instance, an interface target of type `http-api` is additionally described by URL and authentication details.

The *Action* concept is realized by CACAO command objects and forms another integral part. Commands are defined as executable items that contain nothing more than a type and version attribute as well as the (encoded) command itself. Five command types – manual, `http-api`, `ssh`, `bash` and `openc2-json` – are predefined by a CACAO vocabulary and thus cover manual and automated actions. CACAO couples these commands and

targets within workflow steps and requires each command to be executed by all listed targets in the workflow step object. Currently, the CACAO specification does only list a few exemplary commands. The `command` attribute captures string values and it remains open if these are rather technical or organizational in scope.

CACAO does not directly address the *Artifact* concept. Closest to artifacts CACAO defines variables to capture various forms of information relevant for incident response execution. A given variable in CACAO can for example contain a specific IP address upon which a command is performed. Typically, variables are defined on a playbook level but values are used in workflow steps by targets. It is worth mentioning, that at the current point it seems possible that commands will eventually subsume variables. However, there is no further convergence with STIX2.1 CTI objects to provide variable values.

Technological Concepts: Technological concepts are present in CACAO. Next, for CACAO the community, application and serialization concepts are analyzed.

CACAO is developed by an OASIS technical committee supported by multiple large organizations of the information security industry. OASIS further allows interested organizations to participate at the collaborative standard development. Due to its early stage the *Community* concept of CACAO is missing a technical knowledge repository and documentation of implementing CACAO applications. CACAO is licensed according to the OASIS IPR policy and non-assertion mode which allow widely usage.

Technological integration and the *Application* concept is pursued by CACAO through command and target types. Built upon variables possibly taken from other CTI artifacts, CACAO solely directs its commands at a limited number of generic target types. This can be interpreted as direct conversion contained within the format specification. For instance, Application Programming Interface (API) endpoints and Secure Shell (SSH) are two types of more technical targets that might directly use formatted commands. Overall, CACAO is less focused on technical implementation and instead integrates well with organizational processes. Hence, CACAO centers on higher-level incident response standardization.

Serialization of information in CACAO format is based on JSON. JSON is mandatory for implementation but the CACAO specification is defined independently. At the moment no JSON validation schemes for CACAO exist.

Security Concepts: To complement the core concepts of incident response standardization, security concepts and their implementation in CACAO are analyzed below.

The concept of *Confidentiality* is included in CACAO. The fact that data markings have a high significance is reflected by a standalone CACAO object that supports confidentiality. These data markings allow to inform about how to handle and share the described incident response information on a playbook level. TLP with its categories red (named recipients only), amber, green and white (no restrictions) as well as the more extensive Information Exchange Policy (IEP) framework by the Forum of Incident Response and Security Teams (FIRST) are mentioned within the specification. FIRST IEP

2536

IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 23, NO. 4, FOURTH QUARTER 2021

extends TLP by also covering recommendations for encryption and permitted actions. CACAO allows multiple markings to the same playbook but on purpose does not specify rules for their application. Lastly, privacy considerations regarding potential correlation and republication of incident response information are made by CACAO.

The concept of *Authorization* is not implemented by one central CACAO construct. Instead different elements allow forms of authorization. One such element is the impact attribute of playbook objects. The impact value indicates the consequences implied at playbook execution on the organization. An example given by the CACAO specification is the lower impact of investigation compared to remediation tasks. A playbook and its workflow steps can further be tied to organizational processes through the chosen actuator types (e.g., individual or group) or directly by the owner property of workflow steps. Variables then allow the customization according to responsibilities within an organization.

CACAO makes use of playbook object attributes to store information about the *Prioritization* of incident response procedures. A playbook can contain information about its relative priority indicated by a value between 0 and 100. Additional, the severity attribute provides a score for the seriousness of the incident addressed by a given playbook. This implies that security incidents differ in the consequences they have on organizations and thus are of different importance. It must be noted, that eventually the values of these attributes are both subjective and relative. CACAO users must therefore deal with implementing adequate rules to assign comparable values.

CACAO – Summary and Recommendations

- Playbook-centric approach to interorganizational incident response automation with JSON serialization
- Specification backed by well-known industry supporters under OASIS technical committee supervision
- In-depth coverage of most core concepts of incident response standardization and security awareness
- Structural focus on workflows and organizational integration accompanied by multiple (technical) commands
- Missing consideration of CTI integration and vague low-level artifacts of incident response actions
- Ambitious use case definitions with information sharing and digital signing of playbooks
- Additional guidance through best practices for implementation is needed
- Improvements of terminology and naming conventions possible to foster unambiguous semantics throughout CACAO
- CACAO could be considered when searching for a more technical and incident response focused alternative to Business Process Model and Notation (BPMN)
- CACAO could be adopted for SOC/CERT processes and connected with standards of the CTI ecosystem

B. Collaborative Open Playbook Standard (COPS)

Version: Collaborative Open Playbook Standard (COPS) Version 0.2 [38].

Basics: Automation and structured expression of incident response procedures is the overall objective of the *Collaborative Open Playbook Standard (COPS)* data format. Following its inception in 2016, COPS remained closely associated with SOAR software. In many aspects the usage of COPS is tied to the Cortex XSOAR (formerly known as DEMISTO) chat operations platform for incident response and other security tasks. It is at least partly unclear if and how COPS itself is maintained beyond an openly accessible GitHub repository. As for now COPS is not standardized as incident response format by any recognized standardization body. Licensed under MIT license the COPS serialization is based on YAML version 1.2.

Aims: COPS describes an approach to standardize incident response with a format strongly influenced by and tied to a SOAR software product. Pursuing the goal of establishing an open standard for incident response, COPS aims to fully automate incident response playbooks where possible. As another objective, COPS commits itself to enhancing visibility of organizations' incident response procedures. In addition, the exchange of COPS playbooks is considered.

Statistics: As COPS is associated with the Cortex XSOAR software information about the incident response format is mainly extracted from the software documentation as well as the COPS⁵ and Demisto content⁶ GitHub repositories. These constitute the most reliable sources for COPS.

- Peer reviewed academic literature on COPS does not exist. A key word search using the exact terms “Collaborative Open Playbook Standard (COPS)” OR “Demisto playbooks” OR “Demisto COPS” yielded one result in the previously mentioned databases (see Section IV-A). The identified preprint however only briefly describes Demisto and its playbooks [103].
- Gray literature on COPS includes the format specification outlined in the aforementioned GitHub repository [38]. Besides, the Cortex XSOAR developer documentation describes specifics on playbooks and their use [104]. In the Demisto content repository some example playbooks and schemes can be found [105]. Additionally, COPS received some attention from online information security news sites related to its inception in 2016. A published Demisto special edition of *Security Orchestration For Dummies* provides some more useful information about playbooks envisioned to adhere to the COPS format [106].
- Latest developments around COPS are limited. If the surrounding software is considered developments include the change in name of Demisto to Cortex XSOAR by Palo Alto Networks. While Cortex XSOAR is proprietary the COPS format and example content including integrations in other security products remains open-source. The current COPS specification version is 0.2. As of August

⁵<https://github.com/demisto/COPS>

⁶<https://github.com/demisto/content>

2020, playbook schemes have been removed from the content repository.

General Concepts: The analysis of general incident response core concepts shows that for the *Aggregability* concept COPS includes playbooks to document incident response procedures. Playbooks contain individual steps adjusted to a given use case and possible related security product integrations. Different incident types can be specified to trigger a playbook.

Categorization of incident response tasks in COPS does not adhere to the aims of different incident response processes such as investigation or remediation of a security incident. Instead, categorization in COPS is broadly aligned to the categories *manual* and *automated*. This however is of little importance to the objectives of the incident response standardization. Overall, the COPS format covers elements to achieve the generic incidents response aims but does not address these explicitly.

On a structural level, the *Granularity* concept in COPS is realized with playbooks, tasks and commands. Thereby, playbooks express the high-level incident response process. Tasks constitute steps within the process and contain procedural logic. Lastly, specific commands ensure the execution by mentioning precise elements of scripts (e.g., functions) or manual actions.

The concept of *Versioning* is rudimentarily contained in the COPS format. While there are properties to capture version numbers other information such as timestamps about playbook or task creation are missing. In addition, guidance on playbook as well as task creation is brief.

COPS playbooks are identified by a unique ID. The specification mentions UUIDs but no specific UUID version. On playbook level, COPS demands a unique ID for its tasks. *Referencing* and reusing COPS elements without reimplementation is thus possible. Besides, a dedicated task of type *playbook* can be leveraged to refer to another playbook and its procedures. External referencing in COPS is associated with integrations and is relevant to provide context to commands in form of scripts and execution environments.

COPS does not deal with the concept of *Extensibility*. An indirect method to extend COPS is by implementing additional and new integrations for other security tools which can be referenced by COPS task objects. This however is not part of the format specification.

COPS is a clearly technically-centered incident response automation format. With regard to the *Readability* concept it must be stated, that the YAML serialization specification itself claims to be “easily readable by humans”. Nevertheless, human-readability is only given to a certain extent. Analogous to other serializations YAML becomes difficult to comprehend for larger and deeper structured documents. Machine-readability on the other hand is strongly supported by parsing the key-value pairs and the indentation structure of YAML documents with programming languages.

The concept of *Unambiguous Semantics* is only partially addressed by the COPS format. While the specification describes playbook and task properties it is missing data types and further elaborations about the definitions of the incident

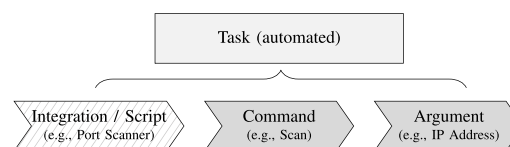


Fig. 7. Structural Description of COPS Playbooks.

response format elements. It further remains unclear why certain information (e.g., type) is redundantly stored in tasks. Whereas the term Digital Forensics Incident Response (DFIR) is sometimes applied to playbooks no specifics on digital forensics are revealed as a terminology section is missing.

Structural Concepts: The analysis of structural concepts reveals the COPS implementation of *Task*, *Integration/Script*, *Command* and *Argument* elements depicted in Figure 7. In the following, definitions as well as parts of the surrounding product ecosystem are analyzed. An exemplary task might utilize a port scanner, its integration as Python script and consist of a scan of an IP address. A noteworthy exception exists for manual tasks which instead of scripts employ human actuators.

The *Workflow* concept in COPS is represented by task objects. These tasks fulfill the need for conditional logic in incident response standardization. Different task types (e.g., start, condition, regular, title, etc.) explicitly deal with conditions, procedural elements and structuring. In general, a task can be distinguished in manual or automated task. This categorization however is not reflected in a specific property but must be inferred from omitted properties (e.g., script). The most granular task – regular task – contains essential information about execution such as integration and script. Besides, tasks store information about following tasks.

In COPS an *Actuator* is a given script of a security product integration. These scripts, mostly written in Python, introduce execution engines. In the case of manual tasks, actuators can also refer to people and processes. Actuators are defined by their name and relate to the respective integration.

The *Action* concept of incident response standardization is defined by command elements. In COPS commands are specific for a given integration and its scripts. Therefore, commands closely resemble function calls with certain input and output values. Through the `is_command` property and its boolean value it is possible to specify if a certain action is directly executable by a script function. Otherwise additional context is needed for execution.

Script arguments provide input values for command execution in COPS. The *Artifact* concept is thus found in COPS. Arguments not only cover objects of incident response as `targets` but also capture variables for the commands as `options`. Hence, it can be observed that this type of structural implementation mixes details on the actual commands with details on artifacts, i.e., objects of command execution.

Technological Concepts: Analysis of the technological *Community* concept shows that COPS is based on a proprietary software product but open-source integrations are collaboratively maintained by an active community. Despite the broad coverage of integrations and scripts for numerous security

2538

IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 23, NO. 4, FOURTH QUARTER 2021

products, there is a serious lack of a detailed specification and maintenance of the COPS format. Information about the format is not only incomplete, but must also be derived from the actual implementations. As the format specification builds the backbone of many practical application aspects, improvement is necessary.

Technological integration of COPS, as the *Application* concept describes, is above all warranted through its use in Cortex XSOAR. Additionally, the integrations emphasize the use cases in which COPS constitutes a connecting element to other relevant security tools. COPS therefore follows an indirect proxy layer approach by maintaining a generic format-based description yet providing specific integrations for individual security tools and actuators.

The *Serialization* of COPS is based on YAML 1.2. COPS uses the indentation structure of YAML to separate the different structural elements. To the best of our knowledge, no schemes exist to assess the adequate serialization of COPS playbooks with regard to data types.

Security Concepts: COPS does not address the *Confidentiality* concept. No properties exist to capture information on confidential data handling such as data markings.

Authorization is an aspect of incidents and playbooks in the Cortex XSOAR solution. In contrast, the COPS format itself does not store information about approval mechanisms for playbook execution. Incident response owners and impact scores are thus not part of this incident response format.

A *Prioritization* concept does not exist for COPS. Playbooks described with COPS might be enhanced with information about the severity of incidents but this is left to implementations using the format. Overall, it must be noted that security concepts cannot be found in the COPS specification.

COPS – Summary and Recommendations

- Playbook-centric approach to incident response automation with YAML serialization and scripts
- Strong technological focus supported by community-driven powerful open-source integrations
- Format and use cases related to proprietary Cortex XSOAR solution
- Missing coverage of security concepts (confidentiality, authorization and prioritization) within the format
- No format maintenance and wider industry support
- Blurry boundaries between the format and technological integrations with security product targeted scripts
- Specification and documentation constitute a major impediment to using COPS as information is unorganized and limited
- COPS (and Cortex XSOAR) could be considered when searching for a familiar and more incident response focused alternative to Ansible playbooks
- COPS could be adopted for integrations with well-known security products and if willing to commit to Cortex XSOAR

C. Integrated Adaptive Cyber Defense (IACD) Framework

Version: Integrated Adaptive Cyber Defense (IACD) Playbooks – A Specification for Defining, Building and Employing Playbooks to Enable Cybersecurity Integration and Automation 2017 [39] and Integrated Adaptive Cyber Defense (IACD) Baseline Reference Architecture Version 1.0 [107].

Basics: Generic incident response standardization with a cyber defense framework and actionable playbooks is the overall objective of the *Integrated Adaptive Cyber Defense (IACD)* data format. Initiated by the Department of Homeland Security (DHS) and the National Security Agency (NSA) in 2014, IACD is maintained by the Johns Hopkins University Applied Physics Laboratory (JHU/APL). No explicit information on standardization and licensing of IACD is available. However, the IACD content is easily available, some documents contain CC BY 4.0 license information and the project's aim is to provide information for customization for individual use cases. Serialization of IACD workflows is based on XML.

Aims: IACD describes an approach to structure incident response with orchestration levels, playbooks and a surrounding reference architecture. IACD playbooks fulfill the objective of aligning organizational security requirements with incident response procedures via BPMN. Further, customization of IACD playbooks and workflows aims to achieve incident response orchestration and automation tailored to organizations and their technical environment. As the IACD reference architecture specifies orchestration service categories (i.e., sensing, sense-making, decision-making and acting) another aim is to provide contextual guidance for incident response playbooks.

Statistics: Information about the IACD incident response format is aggregated on the project website⁷ and includes a specification document and various examples.

- A key word search using the terms “Integrated Adaptive Cyber Defense” OR “IACD” OR “IACD integrate” in common academic literature databases yielded a number of results. We excluded papers from other research fields using the same 4-letter abbreviation. Several papers cover the overall IACD project and its reference architecture [108], [109], [110], [111], [112]. Besides, [42] and [100] mention the IACD approach and playbook format in connection with other incident response formats.
- Gray literature on IACD includes first and foremost the playbook specification [39] and documentation covering the overarching reference architecture [107]. Literature on workflows, orchestration and playbook details provides additional background information [113], [114], [115], [116]. Exemplary IACD playbooks and workflows in the form of BPMN diagrams and XML schemes can be found on the project website.
- Latest developments around IACD include the publication of examples on shareable workflows in the context of IoCs [113]. Some videos of IACD have also recently been posted.

General Concepts: Playbooks in IACD support the *Aggregability* concept of incident response standardization.

⁷<https://www.iacdautomate.org/>

They group incident response elements such as the initiating condition, process steps and an end state as well as best practices, policies and relationships to regulatory requirements. A number of IACD playbooks ranging from rebuilding a server to determining a mitigation action exist.

There is no emphasis on *Categorization* of incident response tasks in IACD and its playbook format. The closest to task categories for incident response actions is the specification of two types of best practices: *Response Options* and *Mitigation Options*. However, this is not an explicitly stated element of the IACD format.

Granularity is addressed by the three IACD orchestration abstraction levels in the form of playbooks, workflows and local instances to implement incident response standardization. Thus both technical as well as non-technical information is part of IACD. The IACD playbook format itself is centered on a higher, non-technical level only. The execution foreseen by local instances is left unspecified.

The IACD playbook format has no *Versioning* concept in place. Metadata and change mechanisms for playbooks and workflows adapted to organization specific needs do not exist. It is mentioned that playbooks can evolve. However, guidance on how changes should be tracked is missing.

Referencing is contained in very limited form within IACD. Beyond the tenet to support the linking of playbooks there is no actual implementation of this type of playbook referencing in the format specification. IACD also does not provide enumerations or vocabularies for a tailored list of example process steps or initiating conditions. Dependent on the modeling tool (e.g., Camunda Modeler), IACD workflows and their elements expressed in XML can have IDs to support identification and referencing. External referencing includes naming of regulatory requirements and industry standards along the IACD playbook in a dedicated section.

In the broadest definition, IACD is extensible. This is because customization is intended on every level of the IACD format and supported by its universal and vague specifications. *Extensibility* in the form of adding certain attribute values or structuring elements is not part of IACD as these remain unspecified.

IACD covers the *Readability* concept. Human-readability is aimed for at the level of playbooks which include incident response process elements aligned to organizational policies. Here, the visualization through BPMN diagrams fosters human-readability explicitly. Additionally, at the level of workflows machine-readability and machine-to-machine communication is addressed by focus on more technical actions and the conversion of BPMN to XML.

The *Unambiguous Semantics* concept is largely absent for IACD as there is no clearly defined terminology. Most notably, redundancies exist for the definition and instantiation of playbooks and workflows. Both share a number of components yet only vary in negligible instantiation aspects. At the end, their differences are not so much between process and technical orientation but mainly stem from granularity. Technical local instances are out of scope of the definitions provided by IACD and available information is very limited. With regard to ambiguity a key element lies in the BPMN diagram modeling by

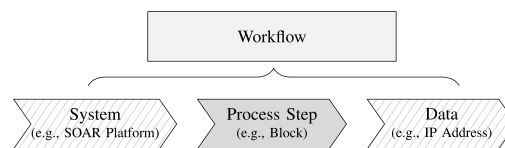


Fig. 8. Structural Description of IACD Playbooks/Workflows.

human analysts which is not addressed by adequate guidance for the incident response automation field.

Structural Concepts: The analysis and representation of general structural concepts in IACD shows a procedural focus. IACD centers on the structural building blocks of *Workflow*, *System*, *Process Step* and *Data* depicted in Figure 8. An exemplary workflow in IACD can involve a SOAR platform to block access to an IP address.

Contrary to other incident response automation formats, IACD workflows can be treated largely independent of IACD playbooks as they are not a component within. The *Workflow* concept is realized by IACD with BPMN diagrams that, analogous to playbooks, contain an initiating condition, process steps and an end state. Structuring of incident response actions is enabled by the conditional elements included in BPMN. With regard to actuators and artifacts it can be derived from the IACD specification that security systems and data eventually represent these concepts. However, it must be stated that workflows still do not warrant a technical implementation of incident response standardization or automation.

The *Actuator* concept is almost entirely absent in IACD. Only textual descriptions along side process steps and overall workflow descriptions hint at information systems and tools used in connection with the described workflows. Whereas BPMN supports the documentation of system-based tasks in the existing IACD examples, no specific actuators are indicated. Extracted from the provided workflow examples, human and system actuator types can be identified.

IACD process steps represent the *Action* concept. As specified by IACD, an incident response procedure is composed of a sequence of documented process steps which are either manually or automatically executed. Each process step is described by its title.

The *Artifact* concept is not part of IACD as it is unspecified. However, the exemplary IACD workflows oftentimes pertain to various forms of IoCs such as IP addresses or file hashes. Artifacts in IACD can thus be found as part of the descriptive process step titles. It is at least arguable if initiating conditions in IACD can also be counted as artifacts.

Technological Concepts: The multitude of IACD documents supports understanding and utilization of the incident response format. The *Community* concept is partially considered as the format specification is non-binding, brief and information on technical implementation is missing. With an active community and U.S. governmental agencies behind IACD, community aspects such as collaboration and maintenance are fulfilled. Participation is further encouraged by IACD events and permissive licensing terms.

Application of IACD is based on the concept of customization. This implies that IACD does not provide any means of

direct conversion. IACD also does not follow a traditional proxy layer approach. Instead, it serves as high-level guidance with its example playbooks and workflows described in BPMN. Incident response standardization is thus entirely dependent on technical interpretation and implementation by organizations. Technical dependencies for IACD are limited to BPMN modeling tools.

The IACD format uses XML *Serialization* for its BPMN workflows.

Security Concepts: The *Confidentiality* concept is missing in IACD. Playbooks and workflows are missing data markings or other means of confidentiality indication.

Authorization in IACD centers on the requirements specified for IACD playbooks. It is defined that besides automation the individual process steps shall reflect human involvement for authorization and approval if necessary.

Severity levels and scoring are not explicitly mentioned within the IACD specification. Thus, *Prioritization* must be introduced when defining and applying IACD playbooks and workflows according to BPMN.

IACD – Summary and Recommendations

- Framework-centric approach to incident response standardization and automation with BPMN diagrams
- Definition of three abstraction levels (playbooks, workflows and local instances) and active community
- Structural focus on process steps and other minimum requirements for playbooks/workflows with extensive examples
- Useful overarching reference architecture for incident response with sensing, sense-making, decision-making and acting
- Missing implementation and incident response emphasis within brief specification documents
- Local instances of workflows and the execution at system level remain unspecified by IACD
- Informal format specification without CTI integration (i.e., artifacts) and unambiguous terminology
- IACD could be considered when searching for a reference architecture to structure multiple incident response formats
- IACD playbooks and workflows could be adopted for generic procedural guidance on incident response actions

D. Open Command and Control (OpenC2)

Version: Open Command and Control (OpenC2) Language Specification Version 1.0 – Committee Specification 02 [40], Open Command and Control (OpenC2) Profile for Stateless Packet Filtering Version 1.0 – Committee Specification 01 [117] and Specification for Transfer of OpenC2 Messages via HTTPS Version 1.0 – Committee Specification 01 [118].

Basics: Incident response standardization focused on machine-to-machine communication is the overall objective of the *Open Command and Control (OpenC2)* data format.

Initiated by the NSA in 2015, OpenC2 was transferred to the nonprofit OASIS. Three subcommittees for the OpenC2 language, OpenC2 implementation considerations and OpenC2 actuator profiles pursue the format development. There exist approved specification documents for the OpenC2 format provided under the non-assertion mode of the OASIS IPR policy. OpenC2 specifies serialization rules for JSON.

Aims: OpenC2 describes an approach to apply command and control mechanisms to cyber defense systems. OpenC2 commands aim to achieve incident response standardization and active cyber defense in a timely manner. The nonproprietary format has the objective of security orchestration and automation independent of the underlying technologies by function-centric interfaces. This includes focus on granular actions, machine execution, transfer of messages and thus the acting part of cyber defense.

Statistics: Among incident response formats, OpenC2 has gained wider attention. Information about OpenC2 can be derived from both the accepted specification and academic literature.

- Peer reviewed academic literature on OpenC2 most notably includes the recently published paper by Mavroeidis and Brule [119], two active supporters of the incident response automation format. In their work the authors provide an extensive description of OpenC2, its concepts, functions and use cases as well as the format’s position within the wider CTI ecosystem. Additionally, the search terms “OpenC2 information security defense” OR “OpenC2 command” OR “OpenC2” applied to common academic literature databases and Google Scholar yield further relevant papers. References [100], [120], [121], [122] and [123] briefly describe OpenC2 or highlight its use within the scope of adjacent research. Applebaum *et al.* [42] emphasize integration of OpenC2 with their proposed playbook specification format RECAST.
- Gray literature on OpenC2 includes numerous news articles about the ideas of OpenC2 and its supporters, listed on the OpenC2 website.⁸ Here, links to various open-source implementations and their code on GitHub can also be found.
- Latest developments around OpenC2 show the proof of concept for integration of various technologies described in recent literature [119]. The newly designed OpenC2 website further encourages participation and use of the incident response format.

General Concepts: OpenC2 is based on defined OpenC2 commands. These short messages contain essential execution information but are not aggregated and arranged in playbooks to document a comprehensive incident response procedure. OpenC2 thus has limited coverage of the *Aggregability* concept. Stated in the language specification, OpenC2 intentionally excludes “sensing, analytics, and selecting appropriate courses of action” and instead centers on the elementary standardization at the technological end [40]. Elements of aggregability can be seen in the content of OpenC2 commands

⁸<https://openc2.org/>

if the level of precision supports multiple OpenC2 actuator or target objects.

Categorization of incident response tasks is not explicitly performed by OpenC2. Analogous to the limited aggregability, the procedural elements of determining an incident response strategy with a specific aim are delegated to prior analysis and organizational processes. Yet, from the different possible actions of OpenC2, to some extent, information about the task categories can be derived. Here, it becomes clear that focus of OpenC2 is on mitigating and remediating existing incidents as well as preventing future ones.

Granularity is achieved by OpenC2 only on a detailed technical level. Information expressed in OpenC2 format is structured according to its use by cyber defense systems. Therefore, various structural elements are defined by properties. Commands, for instance, are further specified by actions. Whereas other incident response formats aim to cover the full incident response spectrum and subsequently often miss technical details, OpenC2 constitutes a format with highly granular objects.

OpenC2 commands contain metadata and thus fulfill the *Versioning* concept. Metadata in OpenC2 includes properties to capture information on the producer, recipient and the creation time of an OpenC2 command. Further, status codes as well as `content_type` (i.e., application/openc2) and `msg_type` (command or response) help to document information associated with the message content. A detailed concept for the information life cycle is out of scope of the OpenC2 format as it is focused on command messages and acknowledgment/response messages only. It can be assumed, that in most cases once an OpenC2 command has been received, interpreted and responded to it becomes outdated. However, referencing allows taking previously issued commands into account. Extensions such as new instances of structural OpenC2 elements are possible and procedures specified in the format documentation.

Referencing is part of OpenC2 and its common message elements. Above all, two unique identifiers are used. As OpenC2 encloses commands in messages, identification is realized by a unique `request_id` part of the metadata and supported by referencing command content with a unique `command_id`. The request identifier should adhere to the UUIDv4 format. For the optional command identifier a 36 character string is specified. Referencing also includes instances of OpenC2 objects. For example, actuator profiles have a unique name and a namespace identifier (NSID). In OpenC2, specifier properties are used to identify a specific actuator or target. External referencing of target objects already part of CTI is not envisioned in the OpenC2 format.

First and foremost, the concept of *Extensibility* in OpenC2 manifest within the extension of actuator profiles. Advancement and introduction of new cyber defense systems can mandate extension of these actuators and their functionality to maintain effectiveness of OpenC2 commands. Precise rules how to introduce new actuator profiles as well as other structural objects include naming conventions and examples. Extensibility is also possible for OpenC2 target objects, command arguments, responses and transfer mechanisms.

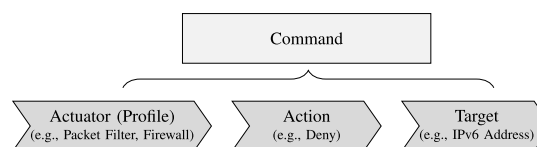


Fig. 9. Structural Description of OpenC2 Messages.

Excluded from extension are the OpenC2 action objects due to the objective of ambiguity avoidance.

Readability of information adhering to the OpenC2 format is based on its description in JSON. Thus machine-readability is warranted. It can further be argued that concise information expressed in OpenC2 messages is comparatively easy to comprehend and fulfills human-readability requirements.

Unambiguous Semantics in OpenC2 is addressed with a terminology section explaining the format's building blocks. In this regard, the format provides a very clear definition of structural components with adequate examples that foster a thorough understanding. Additional, graphical overviews enhance the format specification and document the position of the OpenC2 format in use case scenarios with OpenC2 commands issued by producers and received by consumers. Overlaps with other areas relevant for implementation are highlighted and detailed lists of possible instances for structural components given.

Structural Concepts: The analysis and representation of general structural concepts (*Workflow*, *Actuator*, *Action* and *Artifact*) in OpenC2 shows a technical orientation. OpenC2 centers on the structural building blocks of *Command*, function-centric *Actuator (Profile)*, *Action* and *Target* depicted in Figure 9. An exemplary command in Open2 can employ the stateless packet filtering actuator profile of a firewall to deny access to or from a specific IPv6 address.

As OpenC2 is centered on a message-response system, the *Workflow* concept is represented by the structural component of commands. Command objects form the bracket around actuator (profile), action, and target. An OpenC2 command consists of at least two elements – an action-target pair – as other elements are optional. In OpenC2, generic workflows and conditional logic do not exist and are delegated to prior incident response steps. In contrast to other incident response formats, the granular focus of technical OpenC2 commands emphasizes on the essential incident response action. The OpenC2 command thus clearly defines actions and only supports a defined number of instantiations. All OpenC2 commands support automation and are intended to be handled in an automated way.

The *Actuator* concept in OpenC2 is associated with actuator profiles and covers incident response functions of cyber defense systems. Whereas an OpenC2 actuator is a function of a system, the actuator profile specifies relevant elements of the OpenC2 format specification for this particular function. Currently, OpenC2 has specified only one stateless packet filtering (SLPF) actuator profile. In the specification of the SLPF actuator profile, information on applicable targets and actions can be retrieved from a command matrix (actions \times targets) [117]. Within actuator profiles, specifiers are defined

2542

IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 23, NO. 4, FOURTH QUARTER 2021

to narrow down actions to a specific system or a group of systems.

Actions described by a single verb define the execution operation in OpenC2. In this format, the *Action* concept centers on 20 defined actions ranging from *scan* to *allow* to more complex ones such as *investigate* or *remediate*. For each of these actions, the OpenC2 format provides a description. However, only a limited number are applicable and implemented by the actuator profiles. It is easy to conclude that a firewall endpoint and its stateless packet filtering function will be able to allow access to certain IP addresses but cannot perform an investigation of a file. A single action is a mandatory part of every OpenC2 command. Arguments can be included in the OpenC2 command to define properties related to the action (e.g., *start_time*).

The *Artifact* in OpenC2 is termed target. Besides action, only targets are mandatory elements of OpenC2 commands. As the target is the object of an incident response action it is an evidence-based CTI artifact. In total, 18 target types are specified for the OpenC2 format. Assets (e.g., *device*) as well as network-based (e.g., *ipv6_connection*, *domain_name*) and host-based (e.g., *process*, *file*) elements are possible targets.

Technological Concepts: Starting with the *Community* concept, OpenC2 comprises technological concepts. Organized by OASIS, collaboration in the technical subcommittees and support from many organizations resulted in the OpenC2 format and will advance it in the future if necessary. OpenC2 consists of comprehensive specification documents. It is also part of various prototypical implementations found on GitHub and recently introduced in literature [119]. Libraries for Java and Python are among software to integrate OpenC2. Licensed under OASIS IPR policy, organizations can permissively use the incident response format according to their needs.

OpenC2 is centered on interoperability as it aims to decouple functions of security systems and interfaces. *Application* of the format is possible with both a proxy layer approach and direct transfer to cyber defense systems. For the former integrations rely on a middleware that performs translation and transfer to vendor specific protocols and API endpoints. For the latter standardized interfaces or adapters are needed for cyber defense systems to natively understand OpenC2. The transfer of messages is another aspect of application. The *Specification for Transfer of OpenC2 Messages via HTTPS* [118] addresses this topic in detail. In practical implementations the use of the *Open Data Exchange Layer (OpenDXL)* publish-subscribe message fabric additionally supports OpenC2 message exchange

JSON *Serialization* rules are specified for the OpenC2 format. These requirements determine how OpenC2 data types are encoded. OpenC2 excludes other serialization rules (e.g., XML) but acknowledges their existence. In close relation to serialization, OpenC2 messages are comprised of a header and a body (OpenC2 command) part. This design decision supports the use of common transfer protocols.

Security Concepts: *Confidentiality* is not found in the language specification of the OpenC2 format as a distinct element. Instead, it assigns the handling of confidentiality to

the actual implementations. However, OpenC2 covers confidentiality within its transfer specification and defines HTTPS and TLS usage. It must be noted, that on the technical level of OpenC2 messages, privacy and data markings common to other formats might be of less relevance.

Authorization including ownership, sandboxing and impact assessment of incident response procedures is not part of OpenC2. As decision making must be dealt with prior to issuing OpenC2 commands, it can be derived that it is beyond the scope of OpenC2 to address authorization of the actual incident response action.

OpenC2 does not perform *Prioritization* of incident response actions. In OpenC2 commands, no properties exist to capture urgency information. Presumably OpenC2 orchestrators used as proxies and transferring messages will employ some kind of prioritization or ordering functionality.

OpenC2 – Summary and Recommendations

- Command-centric approach to incident response standardization and automation with JSON serialization
- Established OASIS format with a solid documentation including transfer mechanisms and actuators profiles
- Structural focus on granular and unambiguous execution elements indicating CTI integration
- Recent upswing through sample implementations and academic publication
- Intentional exclusion of conditional logic and procedural integration due to technical orientation
- Dependent on security system vendors or community integrations for direct use or proxy approach
- Missing coverage of security concepts (confidentiality, authorization and prioritization) within the format
- OpenC2 could be considered when searching for a technical, transfer-oriented alternative to shell commands and system configurations
- OpenC2 could be adopted for integration of cyber defense systems at one end of an incident response automation pipeline

E. RE&CT Framework

Version: RE&CT Framework 2020 [41].

Basics: Universal incident response standardization with a stage-action matrix framework and actionable response playbooks is the overall objective of the *RE&CT* data format. Initiated as part of the Atomic Threat Coverage (ATC) project, RE&CT is a community approach started in 2019 and inspired by the MITRE ATT&CK framework. Contribution and maintenance of the format is realized with an openly accessible GitHub repository. Since May 2020, there exists an agreed upon (alpha) version of the RE&CT framework provided under Apache 2.0 License. RE&CT is currently not standardized by any standardization body. The serialization of its components is based on YAML.

Aims: RE&CT describes an approach to categorize incident response actions and build a (visual) knowledge base for incident response procedures. Security incident response playbooks are part of RE&CT and provide structure for multiple response actions. A central use case specified by RE&CT is the development and gap analysis of incident response capabilities in the form of people, processes and technology. RE&CT further aims to achieve incident response automation by its playbook templates which integrate with incident response platforms (e.g., TheHive) and also CTI standards (e.g., STIX). The objective to provide universal incident response guidance yet incorporating actionability is an integral element of the RE&CT format.

Statistics: Information about the RE&CT incident response format can be retrieved from the RE&CT⁹ GitHub repository, the RE&CT documentation and the ATC project.¹⁰

- Peer reviewed academic literature on RE&CT does not exist. A key word search using the terms “RE&CT” OR “RE&CT incident response” in common academic literature databases yielded no relevant results.
- Gray literature on RE&CT includes the format documentation covering the individual framework elements [41]. Exemplary playbooks and utilities concerning RE&CT can be found within the GitHub repository and hint at characteristics as well as utilization aspects of the incident response format [124]. Beyond, the format received some recognition from the security researcher community on Twitter and incident response blogs.
- Latest developments around RE&CT include the publication of the framework in its current form. Participation at the repository further indicates that there is ongoing progress and improvement of the alpha version. While the RE&CT framework structure is static, the individual elements still need specification and additional content. For practical use there exists a RE&CT navigator displaying the entire matrix.¹¹

General Concepts: RE&CT includes incident response playbooks and covers the concept of *Aggregability*. RE&CT playbooks contain incident response actions and elements to support structuring. Playbooks are intended to emphasize on procedures relevant for a specific type of security incident. Currently, there exists a playbook template as well as a possible phishing e-mail playbook.

Categorization is an element of RE&CT represented by the RE&CT stages. All incident response actions are assigned to one of 6 stages ranging from preparation to containment and lastly lessons learned. Incident response automation can thus refer to the RE&CT stages for the aims of a particular response playbook and its tasks. Another RE&CT specific categorization structures incident response actions based on the affected artifacts.

When analyzing technical and non-technical information and the *Granularity* concept for the RE&CT format, the

playbook structure is important. Here, information about the incident response procedure is addressed by a workflow section and listed incident response actions. References and required mitigation systems cover some parts of technical information but lack granularity of more technical CTI.

Versioning and metadata exist in rudimentary form for RE&CT playbooks and incident response actions specified by the framework. Only a `created_date` property captures information about time. Modifications mostly affecting playbooks but also extending to customized actions are documented within the RE&CT format. Authorship is the only other type of metadata relevant for versioning that is part of RE&CT. Whereas other incident response formats provide mechanisms coping with versioning and integration of information, this is missing in RE&CT.

RE&CT playbooks reference defined incident response actions of the framework. Every RE&CT response action is identified by a unique identifier. The concept of *Referencing* and the response action IDs within RE&CT adhere to a custom schema. A prefix of RA for response action is followed by a single digit number to indicate the associated response stage (e.g., containment: 3). Another single digit number refers to the RE&CT specific category (e.g., network: 1). This is followed by an additional sequenced number assigned to each response action. For example, blocking an external domain is referred to by RA3103. Linking other playbooks within a given playbook is possible too, as playbooks contain an ID property with prefix RP and a sequenced number. External references in the form of URLs are included in the RE&CT format and stored within a `references` property.

The RE&CT format does not obstruct the concept of *Extensibility* yet does not explicitly include structured elements for extension. However, from a more general perspective the RE&CT framework and its playbooks are a community approach intended for customization. Thus, adding new response actions or providing further details on existing ones is possible. For RE&CT playbooks there are no restrictions on the granularity of the unstructured workflow section values. The RE&CT framework condensed in the response stage × response action matrix can be perceived as rather static whereas the playbook format leans towards extensibility.

Both forms of *Readability* are part of the overall ATC project and RE&CT. The aim for “actionable analytics” is pursued with human-readability and data provision in Markdown format as well as with machine-readability and YAML files for automatic information processing and execution by incident response platforms. When transformed to TheHive templates or STIX objects, JSON serialization is present.

Unambiguous Semantics is only partially addressed by RE&CT. The documentation and repository describe the individual components of RE&CT but do not cover data types and attribute values. As there exists no clear terminology section with definitions the few example playbooks serve as only reference for RE&CT components. For example, the template playbook lists three possible values (low, medium and high) for a severity attribute which remains unmentioned in the documentation.

⁹<https://github.com/atc-project/atc-react>

¹⁰<https://github.com/atc-project/atomic-threat-coverage>

¹¹<https://atc-project.github.io/react-navigator/>

2544

IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 23, NO. 4, FOURTH QUARTER 2021

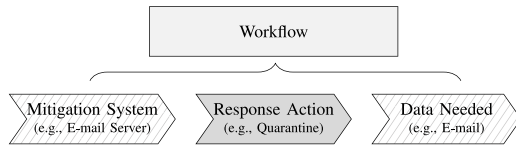


Fig. 10. Structural Description of RE&CT Playbooks.

Structural Concepts: The analysis of structural concepts reveals the RE&CT vision or implementation of *Workflow*, *Mitigation System*, *Response Action* and *Data Needed* elements depicted in Figure 10. In the following, definitions and project content are analyzed. An exemplary workflow might center on an e-mail server to quarantine a malicious e-mail message.

RE&CT playbooks contain a *Workflow* element. Within a RE&CT workflow, there is usually an enumerated list providing instructions in prose on how to execute the relevant response actions for this particular playbook. These response actions themselves are not directly part of the workflow but are structured by response stage listed separately in the playbook. RE&CT workflows aim to address sequential or concurrent ordering of response actions but lack detailed instruments. Derived from the provided exemplary playbook and additional workflow descriptions of individual response actions, it is clear that workflows are intended to foster human understanding of incident response execution.

The *Actuator* concept is represented by the RE&CT vision of mitigation systems. Mitigation systems are not defined as a standalone concept and instead intended to be specified within the *requirements* property. RE&CT pursues the concept of mitigation systems to the point that there are some examples assigned to specific incident response actions. For instance, *MS_dns_server* or *MS_intranet_firewall* document the technical nature of the actuators. Also an automation property links to integration of incident response automation software products. Despite the fact that RE&CT contains a multitude of response actions executed by humans there are no examples of manual actuators found.

RE&CT response actions represent the *Action* concept. As specified in the RE&CT framework, incident response actions align to stages of incident response and can be categorized according to their focal point (e.g., general, network, identity, etc.). In essence, the structure of RE&CT response actions resembles the playbook structure. Every response action is described by its title which contains a single verb and some additional information on the action and the artifact. Throughout, the response actions of the RE&CT framework are more generic and include various combinations of access, analyze, list and find actions. In RE&CT, action and artifact concept are partially merged together. Enforcing a more strict separation of the two concepts could eliminate some of the existing redundancies.

The *Artifact* concept is a placeholder envisioned by RE&CT to be filled with data needed for the incident response action.

Without any information on what characteristics this data holds, it is reasonable to assume two possible directions for implementation. The first direction could include full coverage of the artifact concept by making use of CTI elements. The second direction could focus on explanatory information about how to perform the incident response action only. It should be noted, that to some extent the current RE&CT categories also indicate artifacts. At the end, the structural *Data Needed* concept as part of the *requirements* property is not a standalone object and reflects RE&CT's alpha version.

Technological Concepts: The documentation of RE&CT builds a basis for understanding the incident response format. Nevertheless, the *Community* concept is only partially considered. The format documentation falls short of specifying essential elements in detail. A cohesive list of attribute values and descriptions is missing. In contrast, there is collaboration and a community behind RE&CT. Contributors add content to the repository on GitHub which is under open-source license. This allows adaptation and practical application.

Application of RE&CT follows a twofold approach. Designed as a knowledge base, non-technical application through dissemination of information can be identified. Besides, practical application is realized with a number of provided scripts that directly convert RE&CT content. The content can then be used with other security products. However, generated output (e.g., custom STIX objects) does not always include custom response playbooks and is focused on the RE&CT matrix with its response stages and actions. The RE&CT format serves as an intermediary for incident response automation. Thus, technical application is limited in scope, too.

The RE&CT format uses YAML *Serialization* for its content. No specific YAML version is mentioned and no validation schemes exist. When RE&CT scripts are applied, resulting output (e.g., TheHive templates) is structured according to JSON serialization.

Security Concepts: There are elements of the *Confidentiality* concept present in the RE&CT format. Response playbooks contain a dedicated property for data markings based on TLP. The common TLP scale is applied.

Authorization in RE&CT centers on the Permissible Actions Protocol (PAP) which indicates how received information can be used. Analogous to TLP scale, PAP ranges from white to red with no restrictions on information use, active actions (e.g., block traffic), passive cross check (e.g., third-party services) and up to non-detectable actions only (e.g., local log analysis). Other methods of authorization such as assigning responsibilities and impact assessment are not covered by RE&CT.

Severity levels are captured by a RE&CT property and document consideration of the *Prioritization* concept. The scale for severity indication covers low, medium and high severity of the respective incident response playbook. A more detailed scoring on which response action to conduct first is not given and there are no explanations on the security concepts in the RE&CT format documentation.

RE&CT – Summary and Recommendations

- Framework-centric approach to incident response standardization and automation with YAML playbooks
- Recently started community project transferring the idea behind MITRE ATT&CK to incident response
- Universal knowledge base with scripts to support direct conversion to security products
- Structural focus on incident response actions aligned to stages and RE&CT categories
- Response actions are still incomplete and lack content
- No strict separation of structural components as well as missing details on actuators and artifacts
- Framework character contrary to response playbook (semi-)automation which depends on additional scripts
- Informal format specification without terminology and serialization schemes for validation
- RE&CT could be considered when searching for a familiar and incident response focused alternative to the MITRE ATT&CK framework
- RE&CT could be adopted for guidance and customization of system independent incident response actions

F. Resilient Event Conditions Action System Against Threats (RECAST) Framework

Version: Resilient Event Conditions Action System against Threats (RECAST) Playbook 2018 [42].

Basics: Generic incident response standardization with a framework and incident response playbooks is the overall objective of the *Resilient Event Conditions Action System against Threats (RECAST)* data format. Initiated by the nonprofit MITRE the RECAST project resulted in a playbook specification in 2018. RECAST does not provide any information to aspects of standardization including license and serialization schemes.

Aims: RECAST describes an approach to capture, categorize and automate incident response procedures with a structured playbook format. RECAST incident response playbooks are composed of 14 characteristics and their values. A central use case for RECAST playbooks is to align mission profiles to a subset of plays within a given playbook. RECAST further aims to achieve incident response automation by supporting analysts and reasoning with recommendations. The objective to synthesize important incident response information as well as resilience of course of action decision making are two additional elements of RECAST.

Statistics: Information about the RECAST incident response format is limited to a paper by Applebaum *et al.* [42]. The paper includes the RECAST playbook specification. No gray literature on RECAST exists. As of 2020, it is reasonable to assume that RECAST is deprecated and its development has been discontinued.

General Concepts: RECAST includes incident response playbooks and covers the concept of *Aggregability*. RECAST

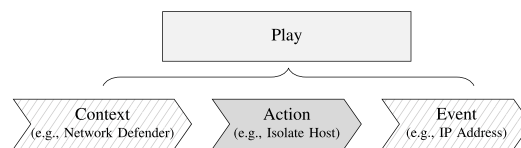


Fig. 11. Structural Description of RECAST Playbooks.

playbooks contain plays and incident response characteristics to support structuring. Alongside mission profiles, playbooks are intended to emphasize on procedures relevant for a specific type of security incident.

Categorization is an element of RECAST and represented by its four categories: events, risks, context and action. These categories however do not reflect incident response tasks. Instead, the *Course of Action Type* characteristic contains information on the incident response task category.

When analyzing technical, non-technical information and the *Granularity* concept, the RECAST playbook specification does not cover detailed technical-oriented information. Plays and actions are the only structuring hierarchies.

Versioning and metadata as well as change mechanisms are not addressed by the RECAST specification.

RECAST playbooks do not incorporate the concept of *Referencing*. From the few provided example plays it can be derived that these plays are eventually identified by a numeric value.

The RECAST format does not obstruct the concept of *Extensibility*, yet does not explicitly include structured elements for extension. Values for specified characteristics are currently based on MITRE internal interview answers.

Readability is part of the RECAST format as incident response information is structured in human-readable prose. In contrast, no measures to support machine-readability are specified.

Unambiguous Semantics is only partially addressed by RECAST. The specification describes the individual components of RECAST but ambiguity is present with playbooks and plays. For instance, it remains unclear if multiple playbooks can exist. Because mission profiles adhere to the playbook structure, their definition is also ambiguous.

Structural Concepts: The analysis of structural concepts reveals that RECAST is based on *Play*, *Context*, *Action* and *Event* elements depicted in Figure 11. An exemplary play might center on a network defender to isolate a host identified from log data with its IP address.

RECAST playbooks incorporate the *Workflow* concept to some extent. Workflows are represented by RECAST plays and include relevant information for incident response in the form of context, action, events, and additional risks. However, the RECAST plays do not contain any information on the conditional logic of executing incident response actions.

The *Actor* concept is represented by the RECAST context category and more specifically the *role* characteristic and its value. As specified, one possible role is the typical user who advocates play execution. Nevertheless, the concept of actuators within the RECAST format remains vague. Systems

commonly representing actuators in other incident response automation formats are not covered by the specification.

RECAST actions fulfill the *Action* concept. Designated *Course of Action* elements capture information on the incident response action. Within RECAST these also contain information of the artifact. It can thus be observed that the action and artifact concept are merged together.

The *Artifact* concept can be identified within the event category of RECAST plays. Event characteristics bundle input information that serves as a trigger for the incident response procedure. Events can also describe the artifact of execution.

Technological Concepts: The documentation of RECAST is limited and aspects of collaboration and the *Community* concept are absent.

Application of RECAST is based on the description of a notional reference architecture. It is envisioned, that a RECAST inference engine and a RECAST responder perform conversion of RECAST plays into executable commands.

The RECAST format does not specify a serialization schema.

Security Concepts: There are no elements of the *Confidentiality* concept present in the RECAST format.

Authorization in RE&CT centers on the generic *Automation Confidence* characteristic. Assignment of automation confidence values implies manual interaction and thus some sort of authorization. Other methods of authorization are the *Role* characteristic for executing the incident response action and the *Consequence* pointing to the impact of incident response.

Risk characteristics can be used to derive information on the severity of incident response actions. Otherwise, the *Prioritization* concept is not addressed by the RECAST format.

RECAST – Summary and Recommendations

- Framework-centric approach to incident response standardization with generic key-value list
- Definition of four information categories (events, risks, context and action)
- Structural focus on playbooks and plays with 14 characteristics of incident response procedures
- Discontinued MITRE project and unused format
- Missing integration of organizational procedures, technical implementation and CTI resources
- Informal format specification with limited examples
- RECAST could be considered when searching for a synthesized, textual description of incident response
- RECAST playbooks and plays could be adopted for human-readable incident response knowledge retention

G. Other Approaches

The incident response formats analyzed and discussed above are complemented by other approaches towards incident response standardization.

1) *General Utility Formats:* The use of general utility formats for incident response standardization and automation is possible to some extent. Despite the fact that these

formats are not unique to application for incident response they provide a number of relevant features. The IT automation tool Ansible [63] can easily be adapted to perform incident response tasks. For this purpose, Ansible requires direct interaction with receiving information systems to enable its ordered task execution. A second general utility format is the Business Process Model and Notation (BPMN) [64]. BPMN is a generic modeling framework for organizational processes and their representation as diagrams. The Open Data Exchange Layer (OpenDXL) format [65] provides a message fabric. Initially tailored to McAfee products its ontology project aims for integration of incident response automation elements. As of now, the ontology specification is still in early stages. Using the Resource-Oriented Lightweight Information Exchange (ROLIE) [66] format for incident response standardization is another option. The IETF RFC 8322 defines ROLIE to support exchange of various types of security information. For the above mentioned general utility formats, integration into an incident response standardization and automation pipeline demands adaptation. Due to missing incident response focus and details we exclude Ansible, BPMN, OpenDXL Ontology, and ROLIE from our detailed analysis.

2) *Digital Forensics Formats:* The digital forensics domain is closely connected to incident response and provides specific data formats to handle forensic data. In particular, digital forensic investigations require data storage and reporting [125]. The Advanced Forensic Format v4 (AFF4) is based on containers to store digital evidence [67], [126]. Analogous, Digital Forensic eXtensible Markup Language (DFXML) has the objective to describe digital forensic information and the results of digital forensic processing [68], [127]. As we separate between digital forensics and incident response, both data formats are beyond the scope of the analysis performed in our paper. Additionally, focus on data storage is similar to elements already present in CTI formats (e.g., STIX2.1 Cyber-observable Objects) [128].

3) *SOAR Products:* Based on two Gartner market guides for Security Orchestration, Automation and Response solutions from 2019 and 2020 we identified SOAR products [35], [129]. The SOAR market has evolved in recent years and there is a multitude of different proprietary products (listed in Table II). These products also incorporate incident response standardization and formats but mostly do not provide any accessible information on specification documents, data schemes and incident response concepts. Whereas information for open-source SOAR products [79], [80], [84], [90], [96] is available, we place our focus of analysis solely on fully specified incident response formats.

V. COMPARATIVE SUMMARY

In this section, we summarize and compare the most important findings of the incident response format analysis from a broader perspective. We first refer back to the categorization used for CTI formats. Then, we highlight analysis results with regard to format usability. General core concepts are briefly discussed. Additionally, we emphasize differences in structural implementation and technological concepts between

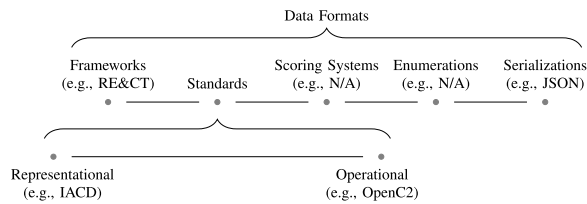


Fig. 12. Categorization of Incident Response Automation Data Formats.

the formats. A comparison of security considerations complements this section. For a complete overview of each format's characteristics associated to the core concepts, we refer to Appendix A Table XI. Likewise, a compact representation of the summary and recommendations for each format is displayed in Appendix B Table XII.

A. Format Categorization

The analyzed data formats share characteristics with existing CTI formats. Therefore, we apply the previously used data format categorization for CTI (see Figure 3) to incident response formats. In Figure 12 categorization of three archetypes of incident response formats is displayed. Inspired by MITRE ATT&CK, RE&CT represents the framework category for incident response. It must be noted that contrary to this categorization, its playbook definition contains elements of the standards category. IACD is the archetypical example of a representational incident response standard. Its BPMN diagrams provide a representational view of incident response processes. OpenC2 is positioned on the other end of the standards spectrum. As an operational standard, it directly concerns the execution of incident response processes. In between this spectrum, the remaining incident response formats RECAST, CACAO, and COPS are located. Scoring systems and enumerations are not present in incident response, but JSON is a typical example of incident response serialization.

An analysis result worth closer consideration is the difference between formats roughly grouped in framework-centric and playbook-centric. The former type always includes a high-level structure and might be further specified on lower levels (e.g., IACD). The latter type does not contain such an overarching framework structure and is, in general, more focused on processes and execution of actions. Indicated by the naming, IACD, RE&CT, and RECAST share framework characteristics. However, playbook elements might also be present in framework-centric formats. Differences between incident response frameworks typically result from additional granular or technological elements. Nevertheless, it can be inferred that framework-centric formats remain broader in scope and contain fewer technical details.

B. Basic Accessibility

Getting acquainted with incident response formats mandates a format specification. Additional information from white and gray literature and the format's recent developments are beneficial, too. A comparison of the analyzed incident response formats with regard to the level of detail of the specification,

TABLE VI
COMPARISON OF SPECIFICATION, LITERATURE AND STATUS

Format	Specification (Detail)	Literature	Status
CACAO	[37] (high)	limited	active
COPS	[38] (low)	limited	inactive
IACD	[39] (medium)	available	active
OpenC2	[40], [117], [118] (high)	available	active
RE&CT	[41] (low)	none	active
RECAST	[42] (low)	none	inactive

the amount of available literature supporting its use, as well as the status is displayed in Table VI. CACAO is characterized by limited additional literature. The COPS format shows a low level of detail for its specification due to missing explanations, structure, and data schemes. Limited literature and COPS' inactive status are deficiencies, too. The IACD playbook specification has a medium level of detail as it is limited in scope. A missing playbook specification proves a low level of detail for RE&CT due to absent data types, schemes, and no explaining literature. At last, limitations of RECAST concerning specification and literature stem from its brief description within a single paper. Further, the RECAST format status is inactive ever since. In the following tables, these deficiencies and other limitations are marked in gray.

C. General Core Concepts

The incident response formats build on general core concepts and use similar methods for their implementation. At this point, one interesting finding concerns the aggregability of information. Here, most incident response formats use playbooks to bundle relevant procedural information. These playbooks reflect the approach pursued by commercial SOAR products. In comparison, the implementation of other general core concepts is more nuanced. Therefore, we refer to the previous analysis and Appendix A Table XI for precise details and side-by-side comparison, respectively.

D. Structural Implementation

Above all, a side-by-side comparison of the individual incident response formats according to the core structural concepts reveals a clear focus on incident response actions. As incident response, in general, is about actively applying countermeasures and performing relevant tasks, the coverage of the action concept by all analyzed incident response formats can be explained. However, the comparison further reveals major weaknesses emphasized in Table VII. CACAO is missing CTI integration as the artifact concept is weakly implemented. For COPS, the strong external dependencies and weak implementation of the actuator concept indicate missing technological integration within the format. Both actuator and artifact concept are unspecified in IACD. Thus, CTI integration and technological integration are absent. OpenC2 is missing organizational integration as the workflow concept is without its scope. Technological integration and CTI integration are missing for RE&CT as both the actuator and the artifact concept show deficiencies. Limitations for RECAST exist

2548

IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 23, NO. 4, FOURTH QUARTER 2021

TABLE VII
COMPARISON OF STRUCTURAL CONCEPT IMPLEMENTATION

Format	Workflow	Actuator	Action	Artifact
CACAO	✓	✓	✓	×
COPS	✓	×	✓	✓
IACD	✓	×	✓	×
OpenC2	×	✓	✓	✓
RE&CT	✓	×	✓	×
RECAST	×	×	✓	×

TABLE VIII
COMPARISON OF TECHNOLOGICAL CONCEPT IMPLEMENTATION

Format	Community	Application	Serialization
CACAO	limited (tech.)	direct/process	JSON
COPS	limited (spec.)	proxy	YAML
IACD	limited (tech.)	process	XML
OpenC2	spec./tech.	direct/proxy	JSON
RE&CT	limited (spec.)	direct	YAML
RECAST	×	proxy	×

for workflow, actuator, and artifact. The reasons behind the structural deficiencies of RECAST are missing CTI integration, imprecise terminology, and limited scope. We conclude that a key element to incident response standardization is to eliminate structural deficiencies of existing formats through extensions or combined use. A combination of representational and operational incident response formats can tackle missing integration and result in a streamlined CTI and incident response environment.

E. Technological Aspects

Differences and similarities between incident response formats and their implementation of technological concepts are displayed in Table VIII. Emphasizing deficiencies, the community concept of CACAO indicates limited technological implementations (e.g., libraries). When applied, CACAO contains direct commands and processes serialized in JSON. COPS has limited community support for its specification and is used as a proxy to different security services. IACD is limited to the technological implementation of BPMN. Its application is process-based and XML serialized. OpenC2 has comprehensive technological and specification implementations, is applied directly or per proxy, and JSON encoded. RE&CT has limited specifications. RE&CT playbooks are directly converted and serialized in YAML. The application of RECAST is proxy-based, but no technical details are known.

F. Security Considerations

Implementation of security concepts in incident response formats varies. Formats either follow strict exclusion, contain no security concepts, or include certain security elements based on considerations relevant to the format's usage. In summary, we indicate in Table IX whether security, in general, is included or excluded.

Security is included in the CACAO incident response format as it covers the concepts of confidentiality, authorization,

TABLE IX
COMPARISON OF SECURITY CONCEPT IMPLEMENTATION

Format	Confidentiality	Authorization	Prioritization	Security
CACAO	✓	✓	✓	included
COPS	×	×	×	excluded
IACD	×	✓	×	excluded
OpenC2	×	×	×	excluded
RE&CT	✓	✓	✓	included
RECAST	×	✓	×	excluded

and prioritization. COPS is missing coverage of security concepts in its format specification. Whereas authorization is partially present in IACD, overall security concepts are absent. OpenC2 explicitly excludes any security concepts and refers to surrounding industry standards for implementation. Within RE&CT, security concepts are included. Contrary, RECAST excludes security concepts but covers parts of the authorization concept. Privacy forms an important topic related to the previously discussed security concepts. Beyond the confidentiality concept and formats, privacy of personal data and regulatory requirements (e.g., EU-GDPR) apply to incident response. We recognize that data formats are limited to fully enforce privacy. Incident response standardization must therefore be accompanied by legal guidelines (e.g., policies) within an organization.

The comparative summary fulfills the purpose of contrasting essential findings. It also aligns with the higher objective of the incident response perspective on CTI to clarify the current status quo of incident response standardization. Relevant meta-information for basic accessibility and valuable outcomes of core concept representation can support decision-making.

VI. INCIDENT RESPONSE STANDARDIZATION USE CASES

Incident response standardization builds the basis for organizational use cases. The format analysis can contribute to assessing incident response use cases and the related identification of the most appropriate standards. In this section, we focus on the three common use cases and arbitrarily defined scenarios for which incident response standardization plays a major role.

A. Automation

A prevalent use case for incident response standardization is automation. Indicated by the analyzed formats' objectives and the multitude of SOAR products and solutions, there is a demand to automate incident response tasks. Tedious and repetitive tasks, as well as swift reaction upon security incidents, cause this development. Further, automation extends existing CTI and embodies the missing incident response perspective.

1) *Scenario*: For automating incident response, we assume a scenario where an organization wants to achieve automated execution of incident response procedures on internal cyber defense systems. The scenario, therefore, includes a strong technical focus as multiple different endpoints (e.g., firewalls and workstations) are involved. Here, incident response

TABLE X
FOCAL POINTS OF USE CASE SCENARIOS

Concept \ Use Case	Automation	Sharing	Reporting
Aggregability	○	++	+
Categorization	○	+	++
Granularity	+	+	○
Versioning	○	++	○
Referencing	+	+	+
Extensibility	○	+	++
Readability	++	+	++
Unambiguous Semantics	+	++	+
Workflow	++	++	+
Actuator	++	○	○
Action	++	++	++
Artifact	++	+	○
Community	+	++	○
Application	++	+	○
Serialization	++	++	○
Confidentiality	○	++	+
Authorization	++	○	○
Prioritization	+	+	○

Legend: ○ less relevant + supporting ++ mandatory

standardization has to cope with integrating existing CTI artifacts on a level that is precise. The automation process flow begins by encoding structured incident response information and transferring it. The receiving system then performs the intended function such as blocking outbound network traffic or removing user privileges.

2) *Core Concepts*: Adapting and using the core concepts for this scenario results in a few focal concepts (see Table X). Above all, structural core concepts have to be fulfilled to enable incident response automation. Due to characteristics of incident response, automated processes must be focused on the detailed description and precise identification of workflows, involved systems (i.e., actuators), the action itself, and the necessary CTI data points (i.e., artifacts). All of these separate accurate from inaccurate incident response. It might be argued that incident response automation will always remain in a semi-automated state as some human involvement is desirable for reasons of accountability and due diligence. Therefore, the authorization concept is emphasized.

Concerning other mandatory concepts, automated incident response is dependent on technical elements. Machine-centered readability, a thorough application concept with technological architecture, and serialization of information are mandatory.

Besides these concepts, a second layer of supporting ones comprises granularity for technical elements, referencing for unique identification, and unambiguous semantics. A supportive community with specifications, reference implementations, and the handling of priorities are also important.

3) *Data Formats*: OpenC2 has a strong focus on structural and technological core concepts. OpenC2 is thus a good fit to support the specified automation scenario. The exclusion of security concepts by OpenC2 is a design choice that must

be considered before implementation. Another suitable incident response format for the automation scenario is CACAO. Despite CACAO's early and currently less technical state, it can cover relevant aspects.

B. Sharing

CTI must be shared among multiple organizations to be most effective. It can be inferred that the same applies to standardized incident response. Disseminated information on incident response procedures supports the common goal of obstructing ongoing attacks and preventing widespread attack campaigns. However, we want to mention that sharing incident response information mandates overarching privacy measures beyond the discussed concepts and formats.

1) *Scenario*: For incident response sharing, we assume a scenario with at least two organizations exchanging incident response procedures. The process flow includes one organization producing structured incident response information and then distributing it over a network to other organizations. The recipients' objective is to apply the received information.

2) *Core Concepts*: The confidentiality concept and the definition of sensitive information are the most important aspects of incident response sharing. Aggregability in playbooks and versioning of information are two other mandatory concepts. Interorganizational sharing further implies a focus on unambiguous semantics as different participants must reach the same conclusion upon the disseminated information. Closer attention is to be paid to workflows and actions as these are relevant from an organizational perspective. A community behind the incident response format is relevant for sharing, as is serialization.

Due to the CTI origin of multiple core concepts, incident response sharing is also supported by several other core concepts.

3) *Data Formats*: More general incident response formats are better suited for the incident response sharing scenario. They typically include confidentiality and have a procedural focus. By assessing coverage of the core concepts, CACAO stands out as one possible candidate due to its comprehensive approach and procedural orientation. In addition, the more generic IACD framework can also be applied as BPMN diagrams provide a universal description easily understandable by multiple organizations.

Incident response formats are not always directly intended for supporting an information-sharing scenario. We point to possible integration with existing CTI formats. In this respect, the STIX2.1 format might be an option to integrate standalone incident response formats via referencing. Hereto, the STIX2.1 Course of Action object will need further details. Consequently, standardized incident response information can be shared without the incident response format fulfilling all requirements for the sharing scenario.

C. Reporting

The reporting use case refers to the documentation of incident response capabilities. Standardized incident response information can support building a dedicated knowledge base

on incident response actions and emphasizing various capabilities within an organization. For that matter, incident response formats go beyond the NIST incident response life cycle and include more detailed capability descriptions.

1) *Scenario*: For incident response capability reporting, we assume a scenario with an organization aiming to document its capabilities in a structured way. Senior management officials receive descriptions of incident response procedures and actions that are implemented on an operational level. For instance, handling of ransomware infections and the preparation aspects of security incidents are covered.

2) *Core Concepts*: Relevant core concepts for the reporting capabilities scenario are, first and foremost, the categorization of tasks within incident response and the action concept. The action concept captures granular information on the precise procedures. Complemented by general core concepts, documentation as the overall objective in this scenario determines extensibility and human-centered readability to be highly relevant.

Supporting concepts range from aggregability to referencing, unambiguous semantics, workflows, and confidentiality. The lower importance of these concepts is based on the internal use within an organization that reduces some requirements.

3) *Data Formats*: Following the focus on categorization and incident response actions, gap analysis indicates the RE&CT framework with its stage-action matrix apt for a reporting capabilities scenario. The framework encompasses RE&CT playbooks to showcase further the transition of incident response capabilities towards actionable playbooks.

VII. CONCLUSION AND FUTURE WORK

The novel incident response perspective on CTI broadens the scope and shifts focus on standardization approaches that outline how to use CTI artifacts for effective cyber defense. In contrast to the prevalent perspectives, the incident response perspective structures CTI artifacts and also adds procedural logic. Our survey introduces core concepts of incident response, assisting efforts to establish and assess different incident response formats. In essence, the few existing incident response formats can be analyzed according to basic information, general, structural, technological, and security concepts. Beyond analysis, incident response core concepts and formats can be leveraged for organizational use cases. These use cases include but are not limited to automation, information sharing, and capability reporting.

As multiple incident response formats and use cases exist, benefits from standardization are manifold. In particular, incident response formats do not only provide added value on their own. Instead, the coupling of multiple incident response formats might prove beneficial for organizations. Together with the integration of existing CTI formats, this can result in a streamlined format system. For instance, an organization using STIX2.1 for generic CTI representation will potentially integrate CACAO for decision-making about incident response workflows and OpenC2 to execute precise incident response actions on defensive information systems. Complemented by

RE&CT's reporting of incident response capabilities, this streamlined format landscape offers a broad basis for many applications.

In this paper, we studied and evaluated existing approaches towards incident response standardization and presented a detailed format analysis. To our knowledge, this is the first comprehensive work to consider incident response standardization and its broad scope of applications. Conclusions drawn from our work base on the following observation: there is a growing interest for structured incident response formats indicated by a surge in SOAR products.

Following these community efforts and cutting-edge developments, we see the necessity for a scientific approach and common understanding. Products and solutions aiming to standardize and automate incident response will rely on underlying data formats that received little attention and are often in their early stages. For existing and yet to be developed incident response formats, an in-depth analysis must be based on a systematic procedure. To this end, we base our study on core concepts of incident response which are partially derived from the encompassing CTI paradigm. The incident response format analysis further reveals that formats center on actions, specific aspects, and do not adhere to the same objective. Therefore, variations in the implementation of core concepts result in deficiencies, strong points, and deem formats more applicable for specific use cases and scenarios than others.

This survey of the incident response perspective on CTI presents a solid foundation for future research. While new standards will emerge, underlying core concepts of incident response are likely to remain the same. However, two aspects warrant a more detailed examination within future work.

- **Privacy** is a very important topic but only partially touches incident response formats (see confidentiality). In contrast, for incident response at large, privacy is a crucial overarching topic. The two reasons why privacy is essential for incident response but barely included in formats are processes and use cases. For some use cases (e.g., sharing), privacy is more important than for others (e.g., reporting). Likewise, processes vary between organizations and require different levels of privacy considerations. Often, privacy must be considered due to legal and regulatory conditions. In addition, organizations will build processes around incident response formats and standards according to their strategic needs. Eventually, these processes and not the formats themselves enforce privacy. We plan future work on the interplay between generic incident response descriptions and organization-specific policies. Adapting information represented in incident response formats will demand research efforts on personal information and privacy-compliant behavior. Interestingly, little is known about incident response policies and privacy compliance measures in incident response so far.
- **Integration and use** of incident response formats on different levels are noteworthy. They will lead to further research – first, the structural concepts of incident response point to CTI artifacts and technologies. Here, future work might address how to extract information

TABLE XI
COMPARATIVE SUMMARY OF INCIDENT RESPONSE FORMAT ANALYSIS RESULTS

Concept \ Format	CACAO	COPS	IACD	OpenC2	RE&CT	RECAST
Aggregability	Playbooks	Playbooks	Playbooks	Limited (commands)	Playbooks	Playbooks
Categorization	Playbook types	N/A	N/A	Limited (actions)	Stages	Limited (CoA type)
Granularity	Workflow steps – commands	Tasks – commands	Workflows – local instances	Commands – actions	Workflows – actions	Plays – actions
Versioning	Metadata; change mechanisms	Limited (metadata)	N/A	Metadata	Limited (metadata)	N/A
Referencing	UUIDv5s; variables	UUIDs; IDs; integrations	Limited (IDs; standards)	UUIDv4s; IDs	IDs	Limited (IDs)
Extensibility	Open vocabularies; STIX2.1 SDOs	N/A	N/A	Actuator profiles; targets	Limited (framework)	N/A
Readability	Machine-centered (JSON)	Machine-centered (YAML)	Human-centered (BPMN)	Machine-centered (JSON)	Machine- & human-centered (YAML/matrix)	Human-centered
Unambiguous Semantics	Limited (definitions)	Limited (data types)	Limited (definitions)	Detailed concept	Limited (data types/definitions)	Limited (definitions)
Workflow	Workflow steps	Tasks	Workflows	Commands	Workflows	Limited (plays)
Actuator	Targets	Limited (integrations)	Limited (system)	Actuator (profiles)	Limited (mitigation systems)	Limited (context)
Action	Commands	Commands	Process steps	Actions	Response actions	Actions
Artifact	Limited (variables)	Arguments	Limited (data)	Targets	Limited (data needed)	Limited (events)
Community	Limited (technical guidance)	Limited (specification)	Limited (technical guidance)	Specification; implementations	Limited (specification)	N/A
Application	Direct conversion; organizational processes	Proxy layer	High-level guidance	Direct conversion; proxy layer	Knowledge base; direct conversion	Proxy layer
Serialization	JSON	YAML	XML	JSON	YAML	N/A
Confidentiality	TLP; FIRST IEP	N/A	N/A	N/A	TLP	N/A
Authorization	Impact; owner	N/A	Limited (human approval)	N/A	PAP	Limited (role/impact)
Prioritization	Priority score; severity	N/A	N/A	N/A	Severity	N/A

from existing formats (e.g., STIX2.1) and connect security systems. Second, existing organizational processes yield valuable information and can be represented by incident response formats. This situation raises questions regarding equivalent representation. Third, CTI formats and incident response formats overlap, and thus redundancy issues might appear. As mentioned, combined format use can be suitable. The use and adaptation of general utility and digital forensics formats excluded from our analysis are also related to integration. Fourth, the use of incident response formats will change, and feedback loops can draw insights from developed libraries, application interfaces, and SOAR products.

We foresee the necessity to follow the ongoing standardization development as this survey documents the current state-of-the-art in early 2021. Continued investigation of privacy, organizational integration, implementation, and compatibility issues of data formats, technologies, and processes are central to fully realize incident response standardization potentials.

APPENDIX A

INCIDENT RESPONSE FORMAT ANALYSIS

See the Tables XI and XII.

APPENDIX B

ACRONYMS

A. Acronym & Description

BPMN	Business Process Model and Notation
CACAO	Collaborative Automated Course of Action Operations
CERT	Computer Emergency and Response Team
CoA	Course of Action
COPS	Collaborative Open Playbook Standard
CSIRT	Computer Security Incident Response Team
CTI	Cyber Threat Intelligence
CAPEC	Common Attack Pattern Enumeration and Classification
CPE	Common Platform Enumeration
CVE	Common Vulnerabilities and Exposures
CVSS	Common Vulnerability Scoring System
CWE	Common Weakness Enumeration
CWSS	Common Weakness Scoring System
ENISA	European Union Agency for Cybersecurity
IDS	Intrusion Detection System
IoC	Indicator of Compromise
IACD	Integrated Adaptive Cyber Defense
IODEF	Incident Object Description Exchange Format
ITIL	Information Technology Infrastructure Library

TABLE XII
HIGH-LEVEL SUMMARY AND RECOMMENDATIONS OF INCIDENT RESPONSE FORMATS

CACAO	COPS	IACD	OpenC2	RE&CT	RECAST
High-level Summary					
Playbook-centric approach to inter-organizational incident response automation with JSON serialization	Playbook-centric approach to incident response automation with YAML serialization and scripts	Framework-centric approach to incident response standardization and automation with BPMN diagrams	Command-centric approach to incident response standardization and automation with JSON serialization	Framework-centric approach to incident response standardization and automation with YAML playbooks	Framework-centric approach to incident response standardization with generic key-value list
Benefits					
Specification backed by well-known industry supporters under OASIS technical committee supervision	Strong technological focus supported by community-driven powerful open-source integrations	Definition of three abstraction levels (playbooks, workflows and local instances) and active community	Established OASIS format with a solid documentation including transfer mechanisms and actuators profiles	Recently started community project transferring the idea behind MITRE ATT&CK to incident response	Definition of four information categories (events, risks, context and action)
In-depth coverage of most core concepts of incident response standardization and security awareness	Format and use cases related to proprietary Cortex XSOAR solution	Structural focus on process steps and other minimum requirements for playbooks/workflows with extensive examples	Structural focus on granular and unambiguous execution elements indicating CTI integration	Universal knowledge base with scripts to support direct conversion to security products	Structural focus on playbooks and plays with 14 characteristics of incident response procedures
Structural focus on workflows and organizational integration accompanied by multiple (technical) commands		Useful overarching reference architecture with sensing, sense-making, decision-making and acting	Recent upswing through sample implementations and academic publication	Structural focus on incident response actions aligned to stages and RE&CT categories	
Shortcomings					
Missing consideration of CTI integration and vague low-level artifacts of incident response actions	Missing coverage of security concepts (confidentiality, authorization and prioritization) within the format	Missing implementation and incident response emphasis within brief specification documents	Intentional exclusion of conditional logic and procedural integration due to technical orientation	Response actions are still incomplete and lack content	Discontinued MITRE project and unused format
Ambitious use case definitions with information sharing and digital signing of playbooks	No format maintenance and wider industry support	Local instances of workflows and the execution at system level remain unspecified by IACD	Dependent on security system vendors or community integrations for direct use or proxy approach	No strict separation of structural components as well as missing details on actuators and artifacts	Missing integration of organizational procedures, technical implementation and CTI resources
Additional guidance through best practices for implementation is needed	Blurry boundaries between the format and technological integrations with security product targeted scripts	Informal format specification without CTI integration (i.e., artifacts) and unambiguous terminology	Missing coverage of security concepts (confidentiality, authorization and prioritization) within the format	Framework character contrary to response playbook (semi-) automation which depends on additional scripts	Informal format specification with limited examples
Improvements of terminology and naming conventions possible to foster unambiguous semantics throughout CACAO	Specification and documentation constitute a major impediment to using COPS as information is unorganized and limited			Informal format specification without terminology and serialization schemes for validation	
Recommendations					
CACAO could be considered when searching for a more technical and incident response focused alternative to Business Process Model and Notation (BPMN)	COPS (and Cortex XSOAR) could be considered when searching for a familiar and more incident response focused alternative to Ansible playbooks	IACD could be considered when searching for a reference architecture to structure multiple incident response formats	OpenC2 could be considered when searching for a technical, transfer-oriented alternative to shell commands and system configurations	RE&CT could be considered when searching for a familiar and incident response focused alternative to the MITRE ATT&CK framework	RECAST could be considered when searching for a synthesized, textual description of incident response
CACAO could be adopted for SOC/CERT processes and connected with standards of the CTI ecosystem	COPS could be adopted for integrations with well-known security products and if willing to commit to Cortex XSOAR	IACD playbooks and workflows could be adopted for generic procedural guidance on incident response actions	OpenC2 could be adopted for integration of cyber defense systems at one end of an incident response automation pipeline	RE&CT could be adopted for guidance and customization of system independent incident response	RECAST playbooks and plays could be adopted for human-readable incident response knowledge retention

JSON	JavaScript Object Notation	OpenC2	Open Command and Control
MISP	Open Source Threat Intelligence Platform	PAP	Permissible Actions Protocol
NCISS	National Cyber Incident Scoring System	PURL	Package Uniform Resource Locator
NIST	National Institute of Standards and Technology	RECAST	Resilient Event Conditions Action System against Threats

SIEM	Security Information and Event Management
SOAR	Security Orchestration, Automation and Response
SOC	Security Operations Center
SPDX	Software Package Data Exchange
STIX	Structured Threat Information eXpression
SWID	Software Identification
TAXII	Trusted Automated eXchange of Indicator Information
TLP	Traffic Light Protocol
TTP	Tactics, Techniques, Procedures
VERIS	The Vocabulary for Event Recording and Incident Sharing
XML	eXtensible Markup Language
YAML	YAML Ain't Markup Language.

REFERENCES

- [1] R. McMillan. (2013). *Definition: Threat Intelligence*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.gartner.com/en/documents/2487216/definition-threat-intelligence>
- [2] D. Chismon and M. Ruks. "Threat intelligence: Collecting, analysing, evaluating." MWR InfoSecurity, Basingstoke, U.K., CERT-U.K., London, U.K., Rep., 2015.
- [3] R. A. Martin, "Making security measurable and manageable," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, San Diego, CA, USA, 2008, pp. 1–9.
- [4] W. Tounsi and H. Rais, "A survey on technical threat intelligence in the age of sophisticated cyber attacks," *Comput. Security*, vol. 72, pp. 212–233, Jan. 2018.
- [5] L. Dandurand *et al.*, "Standards and tools for exchange and processing of actionable information," Eur. Union Agency Netw. Inf. Security (ENISA), Athens, Greece, Rep., 2014.
- [6] J. Steinberger, A. Sperotto, M. Gollinger, and H. Baier, "How to exchange security events? overview and evaluation of formats and protocols," in *Proc. IFIP/IEEE Int. Symp. Integr. Netw. Manage. (IM)*, Ottawa, ON, Canada, 2015, pp. 261–269.
- [7] F. Menges and G. Pernul, "A comparative analysis of incident reporting formats," *Comput. Security*, vol. 73, pp. 87–101, Mar. 2018.
- [8] C. Wagner, A. Dulaunoy, G. Wagener, and A. Iklody, "MISP: The design and implementation of a collaborative threat intelligence sharing platform," in *Proc. ACM Workshop Inf. Sharing Collab. Security (WISCS)*, 2016, pp. 49–56.
- [9] C. Sauerwein, C. Sillaber, A. Mussmann, and R. Brey, "Threat intelligence sharing platforms: An exploratory study of software vendors and research perspectives," in *Proc. 13th Int. Conf. Wirtschaftsinformatik (WI)*, 2017, pp. 837–851.
- [10] F. Menges, B. Putz, and G. Pernul, "DEALER: Decentralized incentives for threat intelligence reporting and exchange," *Int. J. Inf. Security*, vol. 20, no. 5, pp. 741–761, 2021.
- [11] F. Skopik, G. Settanni, and R. Fiedler, "A problem shared is a problem halved: A survey on the dimensions of collective cyber defense through security information sharing," *Comput. Security*, vol. 60, pp. 154–176, Jul. 2016.
- [12] V. Mavroeidis and S. Bromander, "Cyber threat intelligence model: An evaluation of taxonomies, sharing standards, and ontologies within cyber threat intelligence," in *Proc. Eur. Intell. Security Informat. Conf. (EISIC)*, Athens, Greece, 2017, pp. 91–98.
- [13] C. Sillaber, C. Sauerwein, A. Mussmann, and R. Brey, "Data quality challenges and future research directions in threat intelligence sharing practice," in *Proc. ACM Workshop Inf. Sharing Collab. Security*, 2016, pp. 65–70.
- [14] D. Schlette, F. Böhm, M. Caselli, and G. Pernul, "Measuring and visualizing cyber threat intelligence quality," *Int. J. Inf. Security*, vol. 20, pp. 21–38, Mar. 2020.
- [15] V. G. Li *et al.*, "Reading the tea leaves: A comparative analysis of threat intelligence," in *Proc. 28th USENIX Security Symp. (USENIX Security)*, 2019, pp. 851–867.
- [16] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1153–1176, 2nd Quart., 2016.
- [17] J. Zhao, Q. Yan, J. Li, M. Shao, Z. He, and B. Li, "TIMiner: Automatically extracting and analyzing categorized cyber threat intelligence from social data," *Comput. Security*, vol. 95, Aug. 2020, Art. no. 101867.
- [18] A. Berndt and J. Ophoff, "Exploring the value of a cyber threat intelligence function in an organization," in *Proc. IFIP World Conf. Inf. Security Educ.*, 2020, pp. 96–109.
- [19] D. Preuveneers, W. Joosen, J. B. Bernabe, and A. Skarmeta, "Distributed security framework for reliable threat intelligence sharing," *Security Commun. Netw.*, vol. 2020, Aug. 2020, Art. no. 8833765.
- [20] J. D. Howard and T. A. Longstaff, "A common language for computer security incidents," Sandia Nat. Labs, Albuquerque, NM, USA, Rep. SAND98-8667, 1998.
- [21] N. H. Ab Rahman and K.-K. R. Choo, "A survey of information security incident handling in the cloud," *Comput. Security*, vol. 49, pp. 45–69, Mar. 2015.
- [22] C. Islam, M. A. Babar, and S. Nepal, "A multi-vocal review of security orchestration," *ACM Comput. Surveys*, vol. 52, no. 2, pp. 1–45, 2019.
- [23] P. Cichonski, T. Millar, T. Grance, and K. Scarfone, *Computer Security Incident Handling Guide*, vol. 800. Gaithersburg, MD, USA: NIST Spec. Publ., 2012, pp. 1–147.
- [24] C. Alberts, A. Dorofee, G. Killcrece, R. Ruefle, and M. Zajicek, "Defining incident management processes for CSIRTs: A work in progress," Dept. Softw. Eng. Inst., Carnegie Mellon Univ., Pittsburgh, PA, USA, Rep. CMU/SEI-2004-TR-015, 2004.
- [25] J. Van Bon and A. Van der Veen, *Foundations of IT Service Management Based on ITIL*, vol. 3. Zaltbommel, The Netherlands: Van Haren, 2008.
- [26] F. C. Freiling and B. Schwittay, "A common process model for incident response and computer forensics," in *Proc. IT Incident Manage. IT Forensics (IMF)*, 2007, pp. 19–39.
- [27] B. Grobauer and T. Schreck, "Towards incident handling in the cloud: Challenges and approaches," in *Proc. ACM Workshop Cloud Comput. Security Workshop*, 2010, pp. 77–86.
- [28] G. D. Bhatt, "Knowledge management in organizations: Examining the interaction between technologies, techniques, and people," *J. Knowl. Manage.*, vol. 5, pp. 68–75, Mar. 2001.
- [29] B. Schneier, "The future of incident response," *IEEE Security Privacy*, vol. 12, no. 5, p. 96, Sep./Oct. 2014.
- [30] M. Vielberth, F. Böhm, I. Fichtinger, and G. Pernul, "Security operations center: A systematic study and open challenges," *IEEE Access*, vol. 8, pp. 227756–227779, 2020.
- [31] M. J. West-Brown, D. Stikvoort, K.-P. Kossakowski, G. Killcrece, R. Ruefle, and M. Zajicek, "Handbook for computer security incident response teams (CSIRTs)," Dept. Softw. Eng. Inst., Carnegie Mellon Univ., Pittsburgh, PA, USA, Rep. CMU/SEI-2003-HB-002, 2003.
- [32] C. Zimmerman, *Cybersecurity Operations Center*, MITRE Corp., McLean, VA, USA, 2014.
- [33] Executive Office of the President. (2021). *Executive Order 14028 of May 12, 2021—Improving the Nation's Cybersecurity*. Accessed: Sep. 1, 2021. [Online]. Available: <https://www.federalregister.gov/d/2021-10460/p-113>
- [34] M. Bromiley, "Empowering incident response via automation," SANS Inst. InfoSec Read. Room, North Bethesda, MD, USA, White Paper, 2019.
- [35] C. Neiva, C. Lawson, T. Bussa, and G. Sadowski, "2020 market guide for security orchestration, automation and response solutions," Gartner, Stamford, CT, USA, Rep., 2020.
- [36] P. Nespoli, D. Papamartzivanos, F. G. Mármol, and G. Kambourakis, "Optimal countermeasures selection against cyber attacks: A comprehensive survey on reaction frameworks," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 2, pp. 1361–1396, 2nd Quart., 2018.
- [37] OASIS. (2021). *CACAO Security Playbooks Version 1.0—Committee Specification 01*. Accessed: Jan. 15, 2021. [Online]. Available: <https://docs.oasis-open.org/cacao/security-playbooks/v1.0/security-playbooks-v1.0.html>
- [38] DEMISTO. (2018). *COPS—Collaborative Open Playbook Standard*. Accessed: Dec. 15, 2020. [Online]. Available: <https://github.com/demisto/COPS>
- [39] IACD. (2017). *Integrated Adaptive Cyber Defense (IACD) Playbooks—A Specification for Defining, Building and Employing Playbooks to Enable Cybersecurity Integration and Automation*. Accessed: Oct. 15, 2020. [Online]. Available: <https://www.iacdautomate.org/s/IACD-Playbook-Thin-Specification.pdf>

- [40] OASIS. (2020). *Open Command and Control (OpenC2) Language Specification Version 1.0—Committee Specification 02*. Accessed: Nov. 15, 2020. [Online]. Available: <https://docs.oasis-open.org/openc2/oc2ls/v1.0/cs02/oc2ls-v1.0-cs02.html>
- [41] ATC Project. (2020). *RE&CT Framework Documentation*. Accessed: Oct. 1, 2020. [Online]. Available: <https://atc-project.github.io/atc-react/>
- [42] A. Applebaum, S. Johnson, M. Limiero, and M. Smith, "Playbook oriented cyber response," in *Proc. Nat. Cyber Summit (NCS)*, Huntsville, AL, USA, 2018, pp. 8–15.
- [43] M. Liu, Z. Xue, X. He, and J. Chen, "Cyberthreat-intelligence information sharing: Enhancing collaborative security," *IEEE Consum. Electron. Mag.*, vol. 8, no. 3, pp. 17–22, May 2019.
- [44] E. M. Hutchins, M. J. Cloppert, and R. M. Amin, "Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains," *Lead. Issues Inf. Warfare Security Res.*, vol. 1, no. 1, p. 80, 2011.
- [45] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, "MITRE ATT&CK: Design and philosophy," MITRE Corp., McLean, VA, USA, Rep., 2018.
- [46] A. Dulaunoy and A. Ikldoy, "MISP core format," Internet Eng. Task Force (IETF), Fremont, CA, USA, Rep., 2020. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.misp-standard.org/rfc/misp-standard-core.html>
- [47] S. Barnum, "Standardizing cyber threat intelligence information with the structured threat information eXpression (STIX): Version 1.1, revision 1," MITRE, McLean, VA, USA, Rep., 2014. [Online]. Available: <http://stixproject.github.io/getting-started/whitepaper/>
- [48] OASIS Cyber Threat Intelligence (CTI) Technical Committee. (2020). *STIX™ Version 2.1: Committee Specification 01*. Accessed: Oct. 20, 2020. [Online]. Available: <https://docs.oasis-open.org/cti/stix/v2.1/stix-v2.1.html>
- [49] J. Connolly, M. Davidson, and C. Schmidt, "The trusted automated eXchange of indicator information (TAXII)," MITRE, McLean, VA, USA, Rep., 2014. [Online]. Available: <https://taxiiproject.github.io/getting-started/whitepaper/>
- [50] OASIS Cyber Threat Intelligence (CTI) Technical Committee. (2020). *TAXII™ Version 2.1: Committee Specification 01*. Accessed: Oct. 20, 2020. [Online]. Available: <https://docs.oasis-open.org/cti/taxii/v2.1/taxii-v2.1.html>
- [51] Forum of Incident Response and Security Teams (FIRST). (2019). *Common Vulnerability Scoring System Version 3.1: Specification Document—Revision 1*. Accessed: Nov. 1, 2020. [Online]. Available: <https://www.first.org/cvss/specification-document>
- [52] B. A. Cheikes, D. Waltermire, and K. Scarfone, "Common platform enumeration: Naming specification version 2.3," Nat. Inst. Stand. Technol., Gaithersburg, MD, USA, Rep. NIST IR 7695, 2011.
- [53] D. W. Baker, S. M. Christey, W. H. Hill, and D. E. Mann, "The development of a common enumeration of vulnerabilities and exposures," in *Proc. 2nd Int. Workshop Recent Adv. Intrusion Detection*, vol. 7. West Lafayette, IN, USA, 1999, p. 9.
- [54] MITRE. (2020). *Common Weakness Enumeration—A Community-Developed List of Software & Hardware Weakness Types*. Accessed: Nov. 1, 2020. [Online]. Available: <https://cwe.mitre.org/index.html>
- [55] S. Fenz, A. Ekelhart, and E. Weippl, "Semantic potential of existing security advisory standards," in *Proc. 1st Conf. Forum Incident Response Security Teams*, 2008, pp. 1–8.
- [56] J. L. Hernandez-Ardieta, J. E. Tapiador, and G. Suarez-Tangil, "Information sharing models for cooperative cyber defence," in *Proc. 5th Int. Conf. Cyber Conflict (CYCON)*, Tallinn, Estonia, 2013, pp. 1–28.
- [57] L. Dandurand et al., *Standards and Tools for Exchange and Processing of Actionable Information*, Eur. Union Agency Cybersecurity, Athens, Greece, 2014.
- [58] M. Vielberth, F. Menges, and G. Pernul, "Human-as-a-security-sensor for harvesting threat intelligence," *Cybersecurity*, vol. 2, no. 1, pp. 1–15, 2019.
- [59] A. Ramsdale, S. Shiaeles, and N. Kolokotronis, "A comparative analysis of cyber-threat intelligence sources, formats and languages," *Electronics*, vol. 9, no. 5, p. 824, 2020.
- [60] A. de Melo e Silva, J. J. C. Gondim, R. de Oliveira Albuquerque, and L. J. G. Villalba, "A methodology to evaluate standards and platforms within cyber threat intelligence," *Future Internet*, vol. 12, no. 6, p. 108, 2020.
- [61] S. Bauer, D. Fischer, C. Sauerwein, S. Latzel, D. Stelzer, and R. Brey, "Towards an evaluation framework for threat intelligence sharing platforms," in *Proc. 53rd Hawaii Int. Conf. Syst. Sci.*, 2020, pp. 1–10.
- [62] T. D. Wagner, K. Mahbub, E. Palomar, and A. E. Abdallah, "Cyber threat intelligence sharing: Survey and research directions," *Comput. Security*, vol. 87, Nov. 2019, Art. no. 101589.
- [63] Red Hat. (2020). *Ansible—Ansible Is Simple It Automation*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.ansible.com/>
- [64] Object Management Group (OMG). (2011). *Business Process Model and Notation (BPMN) Specification Version 2.0*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.omg.org/spec/BPMN/2.0/PDF>
- [65] McAfee. (2016). *OpenDXL—Open Data Exchange Layer*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.opendxl.com/>
- [66] J. Field, S. Banghart, and D. Waltermire, "Resource-oriented lightweight information exchange (ROLIE)," Internet Eng. Task Force (IETF), RFC 8322, 2018. Accessed: Oct. 20, 2020. [Online]. Available: <https://tools.ietf.org/html/rfc8322>
- [67] M. Cohen and B. Schatz. (2017). *AFF4 Standard V1.0*. Accessed: Oct. 20, 2020. [Online]. Available: <https://github.com/aff4/Standard>
- [68] DFXML Working Group. (2017). *DFXML Schema Version 1.2.0*. [Online]. Available: https://github.com/dfxml-working-group/dfxml_schema
- [69] Micro Focus. (2020). *ArcSight SOAR*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.microfocus.com/en-us/products/arcsight-soar/overview>
- [70] Ayehu. (2020). *Ayehu—Next-Gen IT Automation Platform Powered by AI*. Accessed: Oct. 20, 2020. [Online]. Available: <https://ayehu.com/>
- [71] Palo Alto Networks. (2020). *Cortex XSOAR—Security Orchestration, Automation and Response (SOAR)*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.paloaltonetworks.com/cortex/xsoar>
- [72] D3 Security. (2020). *D3 SOAR—Security Orchestration and Automated Incident Response With MITRE ATT&CK*. Accessed: Oct. 20, 2020. [Online]. Available: <https://d3security.com/>
- [73] Dragos. (2016). *The Dragos Platform*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.dragos.com/platform/>
- [74] EclecticIQ. (2020). *EclecticIQ—Threat Intelligence Powered Cybersecurity*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.eclecticiq.com/>
- [75] Fortinet. (2020). *FortiSOAR—Security Orchestration, Automation and Response (SOAR)*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.fortinet.com/products/fortisoar>
- [76] FireEye. (2020). *Helix Security Platform*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.fireeye.com/products/helix.html>
- [77] DFLabs. (2020). *IncMan SOAR—Automate*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.dflabs.com/>
- [78] Rapid7. (2020). *Security Orchestration and Automation With InsightConnect*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.rapid7.com/products/insightconnect/>
- [79] The Linux Foundation. (2020). *ONAP—Open Network Automation Platform*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.onap.org/>
- [80] Palo Alto Networks-Unit 42. (2020). *Unit 42 Playbook Viewer*. Accessed: Nov. 1, 2020. [Online]. Available: https://pan-unit42.github.io/playbook_viewer/
- [81] IBM Security. (2020). *IBM Resilient Security Orchestration, Automation and Response (SOAR)*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.ibm.com/products/resilient-soar-platform>
- [82] Resolve. (2020). *Resolve—Accelerate Security Incident Response With Automation and Orchestration*. Accessed: Oct. 20, 2020. [Online]. Available: <https://resolve.io/it-automation-resources/accelerate-security-incident-response-with-automation-soar>
- [83] ServiceNow. (2020). *SecOps—Enterprise Security Operations*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.servicenow.com/products/security-operations.html>
- [84] F. Ådegårdstuen. (2020). *Shuffle SOAR*. Accessed: Oct. 20, 2020. [Online]. Available: <https://shuffler.io/>
- [85] Siemplify. (2020). *Siemplify—Security Orchestration, Automation & Response (SOAR) Platform*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.siemplify.co/>
- [86] LogicHub. (2020). *The SOAR+ Platform*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.logichub.com/>
- [87] Honeycomb. (2020). *Honeycomb SOCAutomation*. Accessed: Oct. 20, 2020. [Online]. Available: <https://socautomation.com/>
- [88] Splunk. (2020). *Splunk Phantom Security Orchestration & Automation*. Accessed: Oct. 20, 2020. [Online]. Available: https://www.splunk.com/en_us/software/splunk-security-orchestration-and-automation.html
- [89] Swimlane. (2020). *Swimlane—Security Orchestration, Automation and Response Platform*. Accessed: Oct. 20, 2020. [Online]. Available: <https://swimlane.com/>

- [90] TheHive Project. (2020). *TheHive & Cortex—A 4-in-1 Security Incident Response Platform*. Accessed: Oct. 20, 2020. [Online]. Available: <https://thehive-project.org/>
- [91] ThreatConnect. (2020). *ThreatConnect—Security Orchestration, Automation, and Response Platform*. Accessed: Oct. 20, 2020. [Online]. Available: <https://threatconnect.com/solution/security-orchestration-automation-response/>
- [92] Anomali. (2020). *ThreatStream—Threat Intelligence Platform*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.anomali.com/products/threatstream>
- [93] ThreatQuotient. (2020). *ThreatQ—Threat Intelligence Platform*. Accessed: Oct. 20, 2020. [Online]. Available: <https://www.threatq.com/>
- [94] Tines. (2020). *Tines—Security Orchestration, Automation and Response (SOAR) Platform*. Accessed Oct. 20, 2020. [Online]. Available: <https://www.tines.io/>
- [95] Cyware. (2020). *Virtual Cyber Fusion Solutions*. Accessed: Oct. 20, 2020. [Online]. Available: <https://cyware.com/cyber-fusion-solutions>
- [96] NSA Cybersecurity. (2020). *WALKOFF*. Accessed: Oct. 20, 2020. [Online]. Available: <https://nsacyber.github.io/WALKOFF/>
- [97] P. Clay, “A modern threat response framework,” *Netw. Security*, vol. 2015, no. 4, pp. 5–10, 2015.
- [98] M. Bartock, J. Cichonski, M. Souppaya, M. Smith, G. Witte, and K. Scarfone, “Guide for cybersecurity event recovery,” Nat. Inst. Stand. Technol., Gaithersburg, MD, USA, Rep. NIST SP 800-184, 2016.
- [99] C. Onwubiko and K. Ouazzane, “Soter: A playbook for cybersecurity incident management,” *IEEE Trans. Eng. Manag.*, early access, May 6, 2020, doi: [10.1109/TEM.2020.2979832](https://doi.org/10.1109/TEM.2020.2979832).
- [100] D. Sparrell, “Cyber-safety in healthcare IoT,” in *Proc. ITU Kaleidoscope ICT Health Netw. Stand. Innovat. (ITU K)*, 2019, pp. 1–8.
- [101] A. Harcourt, G. Christou, and S. Simpson, *Global Standard Setting in Internet Governance*. Oxford, U.K.: Oxford Univ. Press, 2020.
- [102] OASIS. (2020). *CACAO Playbook Requirements Version 1.0—Committee Note Draft 01*. Accessed: Nov. 15, 2020. [Online]. Available: <https://docs.oasis-open.org/cacao/playbook-requirements/v1.0/playbook-requirements-v1.0.html>
- [103] R. Puzis, P. Zilberman, and Y. Elovici, “ATHAFI: Agile threat hunting and forensic investigation,” 2020. [Online]. Available: [arXiv:2003.03663](https://arxiv.org/abs/2003.03663).
- [104] Cortex XSOAR. (2020). *Cortex XSOAR Platform Developer Documentation—Playbooks*. Accessed: Dec. 15, 2020. [Online]. Available: <https://xsoar.pan.dev/docs/playbooks/playbooks-overview>
- [105] DEMISTO. (2020). *Cortex XSOAR Platform—Content Repository*. Accessed: Dec. 15, 2020. [Online]. Available: <https://github.com/demisto/content>
- [106] A. Iyer, *Security Orchestration for Dummies: Demisto Special Edition*. Hoboken, NJ, USA: Wiley, 2019.
- [107] Johns Hopkins University Applied Physics Laboratory. (2016). *Integrated Adaptive Cyber Defense (IACD) Baseline Reference Architecture*. Accessed: Oct. 15, 2020. [Online]. Available: <https://www.iacdautomate.org/s/IACD-Baseline-Reference-Architecture-Final-PR.pdf>
- [108] M. J. Herring and K. D. Willett, “Active cyber defense: A Vision for Real-Time Cyber Defense,” *J. Inf. Warfare*, vol. 13, no. 2, pp. 46–55, 2014.
- [109] P. Fonash and P. Schneck, “Cybersecurity: From months to milliseconds,” *Computer*, vol. 48, no. 1, pp. 42–50, 2015.
- [110] K. D. Willett, “Integrated adaptive cyberspace defense: Secure orchestration,” in *Proc. Int. Command Control Res. Technol. Symp. (ICCRTS)*, Annapolis, MD, USA, 2015, pp. 1–13.
- [111] B. K. Döne, K. D. Willett, B. P. Benjamin, D. F. Sterne, G. W. Tally, and D. W. Viel, “Architecting composable security,” *INSIGHT*, vol. 19, no. 2, pp. 58–61, 2016.
- [112] W. Peters, “Integrated adaptive cyber defense: Integration spiral results,” in *Proc. Workshop Autom. Decis. Making Active Cyber Defense*, 2015, p. 1.
- [113] Johns Hopkins University Applied Physics Laboratory. (2020). *Shareable Automation and Orchestration Workflows for Scoring, Sharing, and Responding to Cyber Indicators of Compromise*. Accessed: Oct. 15, 2020. [Online]. Available: <https://www.iacdautomate.org/library>
- [114] Johns Hopkins University Applied Physics Laboratory. (2017). *Integrated Adaptive Cyber Defense (IACD) Orchestration Thin Specification*. Accessed: Oct. 15, 2020. [Online]. Available: <https://www.iacdautomate.org/library>
- [115] IACD. (2017). *Introduction to Integrated Adaptive Cyber Defense (IACD) Playbooks*. Accessed: Oct. 15, 2020. [Online]. Available: <https://www.iacdautomate.org/library>
- [116] IACD. (2017). *Types of Content Within an IACD Playbook[s]*. Accessed: Oct. 15, 2020. [Online]. Available: <https://www.iacdautomate.org/s/IACD-Playbook-Content-Types.pdf>
- [117] OASIS. (2020). *Open Command and Control (OpenC2) Profile for Stateless Packet Filtering Version 1.0—Committee Specification 01*. Accessed: Nov. 15, 2020. [Online]. Available: <https://docs.oasis-open.org/openc2/oc2slpf/v1.0/cs01/oc2slpf-v1.0-cs01.html>
- [118] OASIS. (2020). *Specification for Transfer of OpenC2 Messages via HTTPS Version 1.0—Committee Specification 01*. Accessed: Nov. 15, 2020. [Online]. Available: <https://docs.oasis-open.org/openc2/open-impl-https/v1.0/cs01/open-impl-https-v1.0-cs01.html>
- [119] V. Mavroeidis and J. Brule, “A nonproprietary language for the command and control of cyber defenses—OpenC2,” *Comput. Security*, vol. 97, Oct. 2020, Art. no. 101999.
- [120] N. Kaloudi and J. Li, “The AI-based cyber threat landscape: A survey,” *ACM Comput. Surveys*, vol. 53, no. 1, pp. 1–34, 2020.
- [121] T. Q. Thanh, S. Covaci, and T. Magedanz, “VISECO: An annotated security management framework for 5G,” in *Proc. Int. Conf. Mobile Secure Program. Netw.*, 2018, pp. 251–269.
- [122] N. Nakhla, K. Perrett, and C. McKenzie, “Automated computer network defence using ARMOUR: Mission-oriented decision support and vulnerability mitigation,” in *Proc. Int. Conf. Cyber Situational Awareness Data Anal. Assessment (Cyber SA)*, 2017, pp. 1–8.
- [123] V. Mavroeidis and A. Jøsang, “Data-driven threat hunting using system,” in *Proc. 2nd Int. Conf. Cryptogr. Security Privacy*, 2018, pp. 82–88.
- [124] ATC Project. (2020). *RE&CT Framework Repository*. Accessed: Oct. 1, 2020. [Online]. Available: <https://github.com/atc-project/atc-react>
- [125] M. Stoyanova, Y. Nikoloudakis, S. Panagiotakis, E. Pallis, and E. K. Markakis, “A survey on the Internet of Things (IoT) forensics: Challenges, approaches, and open issues,” *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 1191–1221, 2nd Quart., 2020.
- [126] M. Cohen, S. Garfinkel, and B. Schatz, “Extending the advanced forensic format to accommodate multiple data sources, logical evidence, arbitrary information and forensic workflow,” *Digit. Investig.*, vol. 6, pp. S57–S68, Sep. 2009.
- [127] S. Garfinkel, “Digital forensics XML and the DFXML toolset,” *Digit. Investig.*, vol. 8, nos. 3–4, pp. 161–174, 2012.
- [128] E. Casey, G. Back, and S. Barnum, “Leveraging CyBOXTM to standardize representation and exchange of digital forensic information,” *Digit. Investig.*, vol. 12, pp. S102–S110, Mar. 2015.
- [129] C. Neiva, C. Lawson, T. Bussa, and G. Sadowski, “2019 market guide for security orchestration, automation and response solutions,” Gartner, Stamford, CT, USA, Rep., 2019.



Daniel Schlette received the master’s degree (Hons.) in management information systems from the Elite Graduate Program, University of Regensburg in 2019, where he is currently pursuing the Ph.D. degree with the Chair of Information Systems. Since 2019, he has been a Research Assistant with the Chair of Information Systems. His research interests include the field of cyber threat intelligence. His primary focus within this topic is to leverage structured data formats. The core research results show the importance of data quality aspects, incident response, and threat sharing as application domains.



Marco Caselli received the Ph.D. degree in computer security from the University of Twente with a thesis titled “Intrusion Detection in Networked Control Systems: From System Knowledge to Network Security.” Before starting his Ph.D. he worked with GCSEC, a not-for-profit organization created to advance cyber security in Italy, and Engineering S.p.A., an international company for software development. In 2017, he joined Siemens where he is the Senior Key Expert of the “Attack Detection” topic. His research interests focus on

security of industrial control systems and building automation with a special focus on critical infrastructures.



Günther Pernul (Member, IEEE) received the Diploma and Ph.D. degrees (Hons.) in business informatics from the University of Vienna, Austria. He is currently a Professor with the Department of Information Systems, University of Regensburg, Germany. Previously, he held positions with the University of Duisburg–Essen, Germany; the University of Vienna; the University of Florida, Gainesville; and the College of Computing, Georgia Institute of Technology, Atlanta. His research interests include data and information-security

aspects, data protection and privacy, data analytics, and advanced datacentric applications.

2 Measuring and visualizing cyber threat intelligence quality

Publication information

Current status: Published

Journal: International Journal of Information Security

Date of acceptance: 08 February 2020

Full citation: SCHLETTE, D., BÖHM, F., CASELLI, M., & PERNUL, G. (2021). Measuring and visualizing cyber threat intelligence quality. *International Journal of Information Security*, 20(1), pp. 21-38.

Authors' contributions:	Daniel Schlette	35%
	Fabian Böhm	35%
	Marco Caselli	20%
	Günther Pernul	10%

Journal description: The International Journal of Information Security is an English language periodical on research in information security which offers prompt publication of important technical work, whether theoretical, applicable, or related to implementation.

International Journal of Information Security (2021) 20:21–38
<https://doi.org/10.1007/s10207-020-00490-y>

REGULAR CONTRIBUTION



Measuring and visualizing cyber threat intelligence quality

Daniel Schlette¹ · Fabian Böhm¹ · Marco Caselli² · Günther Pernul¹

Published online: 2 March 2020
© The Author(s) 2020

Abstract

The very raison d'être of cyber threat intelligence (CTI) is to provide meaningful knowledge about cyber security threats. The exchange and collaborative generation of CTI by the means of sharing platforms has proven to be an important aspect of practical application. It is evident to infer that inaccurate, incomplete, or outdated threat intelligence is a major problem as only high-quality CTI can be helpful to detect and defend against cyber attacks. Additionally, while the amount of available CTI is increasing it is not warranted that quality remains unaffected. In conjunction with the increasing number of available CTI, it is thus in the best interest of every stakeholder to be aware of the quality of a CTI artifact. This allows for informed decisions and permits detailed analyses. Our work makes a twofold contribution to the challenge of assessing threat intelligence quality. We first propose a series of relevant quality dimensions and configure metrics to assess the respective dimensions in the context of CTI. In a second step, we showcase the extension of an existing CTI analysis tool to make the quality assessment transparent to security analysts. Furthermore, analysts' subjective perceptions are, where necessary, included in the quality assessment concept.

Keywords Cyber threat intelligence · Threat intelligence sharing · Data quality · Threat intelligence formats · Information security visualization

1 Introduction

The last years have seen the emergence of sharing information about threats, cyber attacks, and incidents by organizations. The urge to join forces in the fight against cyber criminals originates from an ever-increasing number of attacks and the related risks for organizations [1,2]. Not only the number but also the complexity of attacks has increased over the years resulting in successful intrusions with more severe forms of security breaches. For individual organizations, it is an almost impossible task to detect these complex and decentralized attacks on their own. Thus, organizations

share their available information about incidents and attacks. This information is referred to as cyber threat intelligence (CTI).

However, investigations show that inaccurate, incomplete, or outdated threat intelligence is an important challenge for collaborating organizations [3,4]. More recently, empirical studies with domain experts emphasize that ensuring CTI quality throughout the collaboration process is crucial for its continuing success [5,6]. The exchange and utilization of meaningful threat intelligence depends on measuring and ensuring its quality. This necessity is strengthened as the quality of shared information is stated to have an impact on the required time to respond to an incident [7].

Additionally, it is important to inform stakeholders about the quality of individual CTI artifacts [5]. This can help analysts to narrow down available information to the intelligence actually requiring their attention. Therefore, analysts can come to better informed decisions how to react to incidents reported within the CTI. The other way around, the domain knowledge of security analysts is a very promising source for the “fitness for use” [8] of a CTI artifact. Including experts into the process of measuring quality of threat intelligence is a starting point to assess contextually depen-

Fabian Böhm
Fabian.Boehm@ur.de

Daniel Schlette
Daniel.Schlette@ur.de

Marco Caselli
marco.caselli@siemens.com

Günther Pernul
Guenther.Pernul@ur.de

¹ University of Regensburg, Universitätsstr. 31, 93053 Regensburg, Germany

² Siemens AG, Otto-Hahn-Ring 6, 81739 Munich, Germany

dent data quality (DQ) dimensions. To leverage the domain knowledge of experts, it is necessary to make the data quality assessment transparent to them. In a further step, users should be allowed to contribute their own perception of threat intelligence quality which increases the trust into both platform and threat intelligence [9].

This work centers on two aspects making a contribution to measuring cyber threat intelligence quality. We present a first approach to assess relevant quality dimensions of a standardized CTI data format. For this purpose, we first derive relevant DQ dimensions for CTI and define metrics which allow to measure these dimensions. The metrics are then configured to the STIX format as they rely on its structure. We further differentiate metrics which can be calculated automatically and metrics where input of domain experts is needed. Thereupon, we extend our previously proposed open-source CTI analysis tool to convey CTI data quality to security analysts. The extension helps to provide an indication about the quality of the CTI artifact at hand. Our extension also demonstrates how security analysts can contribute to CTI quality assessment through an interactive visualization.

The remainder of this work is structured as follows: Sect. 2 gives an overview of related work in the field of cyber threat intelligence data quality. A brief introduction to the STIX 2 format can be found in Sect. 3. This section additionally provides an example to illustrate the format, the concept of CTI sharing, and related quality issues. In Sect. 4, we select and structure relevant DQ dimensions. Metrics for the assessment of these dimensions in the context of the specific format are configured in Sect. 5. In Sect. 6, we propose an extension of the STIX format for CTI quality and a possible approach to communicate this quality to users of a CTI analysis tool. This section also describes interviews we conducted with security experts to gain feedback on the proposed approach. Our article concludes in Sect. 7 with a short summary and possible future research directions.

2 Related work

Although CTI and especially quality of CTI are not yet extensively researched topics in the information security field, some related work has already been conducted. We give a short overview of this work hereinafter.

Dandurand and Serrano [10] are among the first to define requirements for a CTI sharing platform. The requirements for such a platform include some form of quality assurance and the provision of adjustable quality control processes. The authors, however, do not specify quality dimensions or metrics to assess the quality of the CTI in their proposed infrastructure.

In 2014, Serrano et al. [11] point out that there is missing support for quality control and management in existing CTI

sharing platforms. The authors propose that organizations should install quality control processes to provide multiple measurable quality values. Although the need for quality assessment is discussed, it is not described how such an assessment could be implemented into a platform.

Sillaber et al. [5] perform a series of focus group interviews and discussions with threat intelligence experts. They derive a number of findings on how data quality dimensions influence threat intelligence. They do not identify fundamentally new data quality issues specific to the CTI area. However, the authors give several recommendations for future research and for possibly relevant data quality dimensions. This work does not propose an explicit approach to measure DQ in the CTI context but rather stays on a generic level.

In their survey investigating threat intelligence, Tounsi et al. [7] specifically call for methods to evaluate the quality of threat intelligence. This also applies to the wider organizational security operations center (SOC) context as low-quality CTI is identified to be a pivotal issue [12]. To the best of our knowledge, there is no respective academic work addressing these open issues. Furthermore, none of the currently available commercial threat intelligence sharing platforms is actively measuring CTI quality [7]. With this work, we aim to take a first step into this direction.

3 Structured threat information expression (STIX)

First, this section gives a brief overview of the STIX format. This is necessary as following sections rely on a fundamental understanding of format specifics. The second part introduces a motivational example which is intended to illustrate the STIX format and basic processes of a CTI sharing platform. This example highlights the importance of evaluating CTI quality in the context of a centralized sharing platform with multiple participants.

3.1 STIX format

We base our approach to assess CTI quality on the STIX 2 data format defined and maintained by the OASIS consortium.¹ According to recent analyses, STIX is the de facto standard used for CTI [13,14]. The successor of this format is called STIX 2. It is likely that STIX 2 will reach a similar popularity throughout the next years as it is the format with the most extensive application scenarios [14]. Therefore, our quality assessment is built upon this promising format. Whenever the term “STIX” is used in the remainder of this work, we actually refer to STIX 2.

¹ <https://oasis-open.github.io/cti-documentation/>.

STIX is a machine-readable, semi-structured format based on JavaScript Object Notation (JSON)² to structure and exchange cyber threat intelligence. The format provides two main object types:

1. STIX Domain Objects (SDOs) describing characteristics of an incident and
2. STIX Relationship Objects (SROs) describing the relationships between those characteristics.

SDOs and SROs contain a number of common attributes which are part of any STIX object and additional attributes specific to the respective object type. Common attributes are IDs or the type of the object, whereas exemplary-specific attributes are the motivation of an attacker or the version identifier of a tool.

The current specification of the format conveys twelve SDO types [15]. These allow to provide a holistic view of a cyber incident including both high-level attribution intelligence (e.g., the associated attack campaign or the threat actor) and low-level information (e.g., the data indicating the attack and exploited vulnerabilities).

There are two types of SROs. The first SRO type allows to connect any two SDOs with an explicit relationship highlighting e.g., the vulnerability exploited by a malware. Both can be modeled as SDOs, whereas the logical connection between them is expressed by an SRO. The second SRO type denotes that a specific SDO has been identified. It connects this SDO with an SRO describing the evidential data for this assumption.

SDOs and SROs relevant for a specific threat or incident can be encapsulated by a report. The SDO for this purpose is the *Report* object which references all, respectively, relevant SDOs and SROs.

3.2 Motivational example

In this section, we describe a fictional CTI sharing platform which is used by critical infrastructure providers (e.g., hospitals, energy operators, etc.) to exchange threat intelligence artifacts. Although the platform and the providers in our example are fictional, there is a number of real-world sharing platforms comparable to the described one. The specific characteristics and operation modes of the platform are not relevant to our example which is why we chose a fictional setting. The main goal of the following explanations is to describe the central idea and necessary processes of a CTI sharing platform.

Starting the example depicted in Fig. 1, we can think of a power plant operating a state-of-the-art security operations center (SOC). At some point in time, the alerting mechanisms

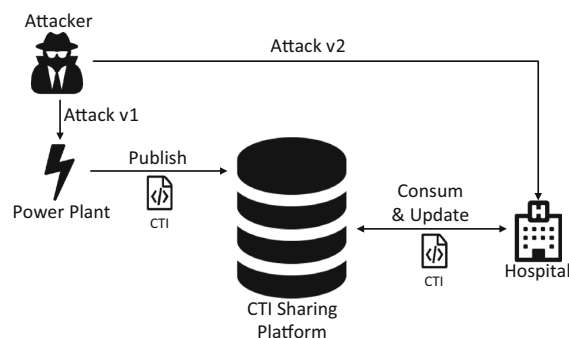


Fig. 1 Simplified CTI Exchange Platform structure

of the plant's intrusion detection systems (IDS) indicate an ongoing attack affecting various critical systems. Automated systems start the collection of related information through log file and network traffic analyses. Immediately, security experts start their analysis to protect the plant's cyber systems and to gain as much insight into the attack as possible.

The outcome of automated and manual analyses in the form of collected, attack-related data casts a light on what seems to be an unknown APT. Various machines of the power plant have been compromised and connected to several control units outside of the internal network. The related IP addresses as well as configuration files have been identified. Additionally, the attackers exploited known but unpatched vulnerabilities of a web server and a specific version of an operating system to spread their attack. This allowed them to conduct lateral movement in the organization's network without being noticed. To defend the network and remove the malware, security analysts applied appropriate countermeasures.

Part of the power plant's SOC is the active participation on a CTI sharing platform. On this platform, several operators of critical infrastructure collaborate to improve their cyber defense. Most of these collaborative efforts are based on exchanging intelligence about previously unknown threats or by sharing new insights about existing incidents. There are different roles of participants active on the platform: Publishers post CTI artifacts on the platform, whereas consumers process these artifacts. However, participants of a sharing platform usually hold both these roles simultaneously.

As the power plant's analysts did detect a new type of attack, they transform the gained insights into a STIX report which is published on the sharing platform. The CTI contains the identified threat actor, exploited vulnerabilities, and the deployed malware. Additionally, the analysts include indicators of compromise (file hashes, IP addresses, and the like) to help other participants to detect this attack. They also share the applied countermeasures.

² <https://www.json.org/>.

A simplified example of the STIX artifact shared by the power plant is shown in Listing 1. Please note that some aspects of the example are not fully aligned with the current STIX specification due to readability reasons.³ However, the example allows to gain a better understanding of STIX. The shared CTI contains the identified *Threat Actor*, the deployed *Malware*, the exploited *Vulnerability*, and an *Indicator* referring to the respective malware file. Additionally, the *Relationships* between these entities are shown. For example, these relationships point out that the *Threat Actor* uses the *Malware* to target a *Vulnerability*.

Another user of the CTI sharing platform might be the operator of a hospital. The operator is leveraging the knowledge made available on the platform to improve the hospital's resilience to cyber attacks. Therefore, published indicators of attacks from the platform are automatically fed into the operator's intrusion detection systems. Additionally, security experts of the operator carry out manual analyses on the most relevant CTI artifacts to identify possible threats. The manual analysis of the artifacts is performed through a visual interface as the CTI format used by the platform is not easily readable for humans.

```
{
  ``type``: ``threat-actor``,
  ``id``: ``threat-actor--1``,
  ``created``: ``2019-04-07T14:22:14Z``,
  ``modified``: ``2019-04-07T14:22:14Z``,
  ``name``: ``Adversary Bravo``,
  ``description``: ``Is known to
    manipulate critical
    infrastructures, I suppose``,
  ``labels``: [ ``spy``, ``criminal`` ]
}, {
  ``type``: ``malware``,
  ``id``: ``malware--1``,
  ``created``: ``2019-04-07T14:22:14z``,
  ``modified``: ``2019-04-07T14:22:14Z``,
  ``name``: ``Malware d1c6``,
}, {
  ``type``: ``vulnerability``,
  ``id``: ``vulnerability--1``,
  ``created``: ``2019-04-07T14:22:14z``,
  ``modified``: ``2019-03-07T14:22:14z``,
  ``name``: ``A Webserver Vulnerability``
}, {
  ``type``: ``indicator``,
  ``id``: ``indicator--1``,
  ``created``: ``2019-04-07T14:22:14Z``
}, {
  ``type``: ``relationship``,
  ``id``: ``relationship--1``,
  ``created``: ``2019-04-07T14:22:14Z``,
  ``modified``: ``2019-04-07T14:22:14Z``,
  ``source_ref``: ``threat-actor--1``,
  ``target_ref``: ``malware--1``,
  ``relationship_type``: ``uses``
}, {
  ``type``: ``relationship``,
  ``id``: ``relationship--2``,
  ``created``: ``2019-04-07T14:22:14Z``,
  ``modified``: ``2019-04-07T14:22:14Z``,
  ``source_ref``: ``indicator--1``,
  ``target_ref``: ``malware--1``,
  ``relationship_type``: ``indicates``
}, {
  ``type``: ``relationship``,
  ``id``: ``relationship--3``,
  ``created``: ``2019-04-07T14:22:14Z``,
  ``modified``: ``2019-04-07T14:22:14Z``,
  ``source_ref``: ``malware--1``,
  ``target_ref``: ``vulnerability--2``,
  ``relationship_type``: ``targets``
}
```

³ Object IDs are not in UUIDv4 format, and some mandatory schema structures are left out.

```
``modified``: ``2019-04-07T14:22:14Z``
``,
``labels``: [ ``malicious-activity`` ],
``pattern``: ``[ file:hashes.'SHA
-256' =
'4bac27393bdd9777ce02453256c5577c
d02275510b2227f473d03f533924f877
']``
``,
``valid_from``: ``2019-04-07T14:22:14
Z``
}], {
  ``type``: ``relationship``,
  ``id``: ``relationship--1``,
  ``created``: ``2019-04-07T14:22:14Z``,
  ``modified``: ``2019-04-07T14:22:14Z``,
  ``source_ref``: ``threat-actor--1``,
  ``target_ref``: ``malware--1``,
  ``relationship_type``: ``uses``
}, {
  ``type``: ``relationship``,
  ``id``: ``relationship--2``,
  ``created``: ``2019-04-07T14:22:14Z``,
  ``modified``: ``2019-04-07T14:22:14Z``,
  ``source_ref``: ``indicator--1``,
  ``target_ref``: ``malware--1``,
  ``relationship_type``: ``indicates``
}, {
  ``type``: ``relationship``,
  ``id``: ``relationship--3``,
  ``created``: ``2019-04-07T14:22:14Z``,
  ``modified``: ``2019-04-07T14:22:14Z``,
  ``source_ref``: ``malware--1``,
  ``target_ref``: ``vulnerability--2``,
  ``relationship_type``: ``targets``
}
```

Listing 1 Exemplary STIX 2 artifact

The power plant's CTI artifact is analyzed by the hospital's security personnel only a few months after the respective incident. This is mainly because vast amounts of available CTI hinder the security experts to identify threat intelligence relevant for them. During the analysis of the artifact published by the power plant, the responsible security analyst of the hospital spots that the respective attack targets a software in use by the hospital as well. Subsequent network and endpoint analyses indicate that the hospital has been affected although the IDS seems to have not noticed the compromise as the binaries of the malware have changed in the meantime. In addition, although the same software is in use, the version number proclaimed to be exploited at the power plant seems to be invalid.

During the analysis of the incident at the hospital, analysts come across some changes and additional insights into the attack. Additionally, the proposed countermeasures are not sufficient to get rid of the attacker. Therefore, an updated ver-

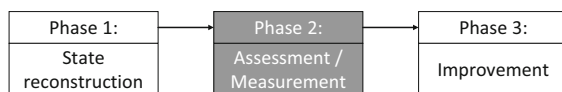


Fig. 2 Process steps of DQ methodologies [16]

sion of the CTI artifact is published to the platform to ensure each participant is informed about the advanced version of the cyber attack. However, during this process the information about the threat actor is unintentionally duplicated leading to redundant information.

The example above shows that the timely exchange of high-quality CTI is crucial for the effort of organizations to prevent cyber security breaches. However, there are numerous pitfalls regarding the quality of the shared threat intelligence. Examples from the above-described use case are: 1) inaccurate information caused by input errors made during the documentation of an attack (invalid version of exploited software), 2) outdated information caused by delays in CTI propagation (changed binaries of malware), or 3) duplicated information caused by collaboration (redundant description of threat actor). Even the overload of CTI available to human analysts and their incapability to determine the most relevant CTI can be seen as a data quality problem. Each of these examples stresses the urge to measure CTI quality and to visualize the results for human analysts.

4 Approach for CTI quality assessment

General DQ methodologies consist of three main process steps depicted in Fig. 2. Initially, the collection of necessary data is performed. Data sources and involved costs are fundamental building blocks for the following process steps. The second step includes the identification and measurement of relevant quality dimensions in the context where the methodology is applied. After quantifying data quality, the last process step strives to improve the quality following a fixed set of techniques and strategies. Although there is no cohesive methodology for information quality management of CTI yet, this work solely focuses on measuring DQ in the context of CTI as highlighted in Fig. 2. Up to now, existing work has mostly provided general advice for mainly the first and the last methodology step but has not described approaches to actually measure CTI quality [5,13,17]. We put explicit focus on the quality assessment. We thus assume the existence and availability of the necessary data for assessment.

Our work on selecting and structuring DQ dimensions relevant for CTI is the result of an iterative process in which we actively sought input and feedback from a number of CTI researchers and practitioners, e.g., domain experts from

computer emergency response teams (CERTs). Throughout multiple evaluation iterations the relevant dimensions and their structure as described in the following two subsections were consequently adapted according to the input of the experts.

4.1 Selecting relevant DQ dimensions for CTI

Before introducing measurements for CTI quality, relevant DQ dimensions have to be selected. Extant work has already suggested a wide variety of different DQ dimensions referring to either the data values or the data schema [18]. The literature distinguishes three main approaches for proposing general and abstract quality dimensions: the theoretical approach [19], the empirical approach [20], and the intuitive approach [21].

Considering the different approaches and various DQ dimensions, it is not an easy task to select relevant and applicable dimensions for a problem at hand. Following the empirical approach by Wang and Strong [20], related research such as the work of Sillaber et al. [5], Sauerwein et al. [13], or Umbrich et al. [22] identify a first set of relevant dimensions which is refined throughout this work.

Our resulting set of dimensions is shown in Fig. 3. An interesting finding yielding from the discussion with the CTI experts is the high complexity of the *Appropriate amount of data* quality dimension. This dimension is meant to help experts to decide whether a CTI artifact by any chance could contain helpful information. In general, this decision can only be made by comparing the real-world artifact with its CTI description. However, this is rarely possible. Therefore, another approach is needed to give security analysts an indication for this dimension. Throughout our discussions, it turned out that experts are often basing their decision on the diversity of SDO types and their interconnection in a STIX report. Arguably, homogeneous SDO types and few relationships between them lead to experts' perception that the report does not describe the real-world incident properly. For the in-depth examination of the *Appropriate amount of data* quality dimension we refer to Sect. 5.3.

4.2 Structuring DQ dimensions for CTI

Our goal for DQ assessment in the context of CTI is to come up with measures to quantify the selected dimensions and aggregate them into a combined score for a STIX report. We therefore structure the dimensions in three different levels depending on the input data as shown in Fig. 3. The assessment of the dimensions on the "Attribute Level" operates on specific attributes of STIX objects, e.g., the dimension of *Timeliness* can be assessed using the *modified-* and *created-* attributes of STIX objects. The two dimensions located on the "Object Level" in Fig. 3 are not bound to predefined attributes

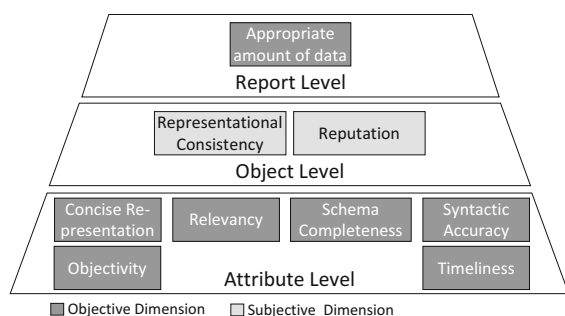


Fig. 3 Schematic of the structure of DQ dimensions

of the objects. In fact, they can be measured based on either varying attribute sets (*Representational Consistency*) or the object as a whole (*Reputation*). At the highest level (“Report Level”), we propose a final dimension to cope with experts’ requirement to be informed about whether a report is likely to contain an *Appropriate amount of data* as described in the paragraph above.

Individual scores on both attribute and object level are then aggregated to a combined object quality indicator. This aggregation provides a quick and helpful insight for any user navigating through cyber threat information. Artifacts with a high-quality score are probably the ones to analyze first. Additionally, on a “Report Level” this aggregation allows to inform users about the average object quality in a given report. This is accompanied by an indication whether the report contains an appropriate amount of data. However, as DQ dimensions can be of varying importance for different users the aggregation has to be customizable [11]. Adjustable aggregation parameters enable CTI users to define the weight of each DQ dimensions in the procedure of calculating a quality indication for each STIX object. The corresponding metrics for aggregation are further outlined in Sect. 5.4.

Additionally, to these various levels for the DQ dimensions, we differentiate objective and subjective dimensions which are also indicated in Fig. 3. DQ has to be evaluated with objective measurements as well as from subjective perception [16,23,24]. Objective measurements rely on mathematical and quantitative measures to evaluate an artifact’s quality. However, some dimensions of DQ are dependent on their contextual environment. It is thus necessary to incorporate the requirements, experience, and knowledge of domain experts. When it comes to the decision whether data is of high quality regarding a specific use case or context, objective DQ dimensions fail to provide reasonable quality scores [25]. At this point, it is necessary to incorporate subjective measures as a supporting concept. Here, the assessment of an artifact’s quality is based on qualitative evaluation by data administrators and other experts.

In the context of a CTI sharing platform, the concept of subjective perception and domain knowledge to evaluate various DQ dimensions equally applies. While domain knowledge is a necessary input for subjective quality dimensions, it also supports assessment of objective DQ dimensions. The domain knowledge can be captured through a system similar to a reputation system where users provide their perception about the quality of an object or report [26]. The need for a reputation system to include subjective quality perceptions and to increase trust is also highlighted in empirical studies [5]. Subjective quality assessment in the CTI sharing context can originate from different stakeholders of a respective platform: On the one hand, consumers (security experts, analysts, etc.) contribute with their domain knowledge and their organization-specific background; on the other hand, a platform host can act as a trusted third party contributing to the quality assessment.

Overall, these three levels provide good and transparent indicators for the quality of a STIX-based cyber threat intelligence artifact. For indication of individual DQ dimensions, we adopt and extend existing naming conventions [20].

5 Measuring CTI quality

In this section, we elaborate on suitable DQ dimensions as the result of our studies. For each dimension, its applicability to the CTI context is described and respective metrics for assessment are configured. Those assessments are either of an objective or a subjective nature depending on whether they can be automated or need manual input. Subjective metrics are based on the perceptions and expressions of a CTI sharing platform’s participants. Furthermore, there is a number of objective dimensions which benefit from additional manual input of domain experts. The ordering of the metrics follows the previously outlined structure of the dimensions in Sect. 4.2.

The proposed metrics in the following are again the result of an iterative process collaborating with CTI researchers and practitioners. Several metric configurations result from long discussions with domain experts where a lot of very valuable feedback was provided highlighting possible configurations to assess CTI quality.

Configurations for the metrics are based on the formal ground truth defined in Eqs. 1–5. We formally define two different attribute sets of STIX as A_r (i.e., Eq. 1) and A_o (i.e., Eq. 2). Required attributes a_r , for example, are unique IDs, names, labels, and types which are present in most STIX objects. As for optional attributes a_o , characteristics such as descriptions, versions, and external references are referred to Eq. 3 which defines any STIX Domain Object or STIX Relationship Object as a specific subset of both the available required and optional attributes. This subsequently allows us

to describe the objects O held by a CTI sharing platform as a set of objects where each object o is either a SDO or SRO (i.e., Eq. 4). STIX objects such as *Threat Actor*, *Malware*, or *Indicator* belong to the set of SDOs, while *Relationship* and *Sighting* objects are SROs. When an incident or an attack is reported to the platform, the resulting report r is defined by Eq. 5 to be a subset of all objects persisted in the platform.

$$A_r = \{a_r \mid a_r \text{ required in STIX 2}\} \quad (1)$$

$$A_o = \{a_o \mid a_o \text{ optional in STIX 2}\} \quad (2)$$

$$SDO, SRO \subseteq (A_r \cup A_o) \quad (3)$$

$$O = \{o \mid o \in (SDO \cup SRO)\} \quad (4)$$

$$R = \{r \subseteq O\} \quad (5)$$

5.1 Attribute level

This subsection defines the DQ dimensions we consider to be assessed at the attribute level, meaning that they rely on a subset of a STIX object's attributes.

Concise representation Concise representation addresses expressiveness of CTI and redundancies within the data [20]. Intensional and extensional are two distinct forms of conciseness. While the former is centered on the uniqueness of attributes and the schema level, the later emphasizes on unique objects. In the motivational example, the duplication of the information about the attacker links to the concise representation dimension as DQ is affected. It is worth noting that in the extant literature, concise representation sometimes only refers to compactly represented information [18]. In the context of STIX, the specification provides clear guidance how to implement a concise representation. It is explicitly stated that a unique identifier is assigned to each artifact. Additionally, each STIX object adheres to a specified JSON schema, and thus, optional and mandatory attributes are predefined. In general, the assumption holds that intensional conciseness is warranted through the schema definition. One exception in STIX is based on specifics of several STIX objects⁴ as they contain lists referencing other objects. These lists are prone to redundant inputs, especially when defined manually.

With regard to extensional conciseness, the information within a CTI platform must be assessed for its respective quality. The main reason for this is that with a growing number of CTI producers, the probability of duplicated objects within the platform becomes likely. More precisely, there is a high chance that two or more objects on the platform are semantic duplicates. Even considering one single STIX report, semantically unique objects are not guaranteed as more than one person could work on the documentation

⁴ Examples are the *Report* object as well as the *Sighting* object.

of the incident and already existing information might be overlooked. Especially, when taking a look at the numerous free-text description fields defined in the current STIX specification, an indication whether these descriptions contain redundant information is important. However, comparing text for semantic redundancy is not an easy task. We encourage the application of methods for semantic similarity. The *Simhash* algorithm is one example proposed to approach this problem [27]. It allows for comparing two STIX objects regarding their uniqueness. An object o_1 is considered unique in a set of objects O if its *similarity* to any other object $o_2 \in O$ is below a threshold t (see Eq. 6).

Objective metrics alone are not sufficient to assess concise representation in practical use. It is inevitable to include subjective perceptions through the utilization of domain knowledge. In this case, platform users conduct or support quality assessment and contribute by pointing out redundancies.

$$CR(o) = \begin{cases} 1 & \text{if } \textit{similarity}(o_1, o_2) < t \\ 0 & \text{else} \end{cases} \quad (6)$$

Objectivity CTI is oftentimes created by multiple human actors during the analysis of an attack. These human CTI creators contribute not only objective threat information but might also introduce emotional or subjective perceptions. Most of the resulting descriptions are phrased in natural language. This is also the case in the motivational example in Sect. 3.2 and the threat actor description. There, the words "I suppose" indicate subjectivity and the context-dependent observations of the security analyst. However, objectivity is a desirable characteristic of shared CTI artifacts as only objective information can be helpful for others. Natural language processing and sentiment analysis, therefore, can facilitate the assessment of unbiased and impartial CTI information as part of the objectivity DQ dimension.

Subjective descriptions of CTI information can be identified through the use of various subjectivity detection methods [28]. In the context of CTI and with regard to STIX, special focus is on attributes with free-text description fields in contrast to predefined enumerations and open vocabularies. This ultimately leads toward a sentence-level orientation for subjectivity detection as these fields contain only a limited number of words. Subjectivity detection methods in general can follow a syntactical approach or center on semantics. A thorough investigation into specifics of such methods must be considered during implementation to determine the best-fitting approach. Regardless of implementation, we classify relevant attribute values $v(a)$ of STIX objects into two distinct categories objective and subjective as shown in Eq. 7. Underlying this classification is the application of a suitable sentiment algorithm which yields a score for either objectivity or subjectivity. The results of the classification for chosen

attribute values are then aggregated to provide an objectivity metric for each object o based on Eq. 8.

$$OB(a) = \begin{cases} 1 & \text{if } v(a) \text{ classified as } \textit{objective} \\ 0 & \text{if } v(a) \text{ classified as } \textit{subjective} \end{cases} \quad (7)$$

$$OB(o) = \frac{\sum_{a \in o} OB(a)}{|o \cap (A_r \cup A_o)|} \quad (8)$$

Relevancy Relevancy forms a DQ dimension incorporating a user's perspective by comparing sets of property values to assess the usefulness of a CTI artifact for the consumer. This is an important aspect of CTI's fitness for use regarding an individual organization or analyst. For example, CTI describing an incident targeted at a specific industry sector is likely to be less relevant for other industry sectors. Also, security analysts might not be interested in threats targeting technologies not deployed in their organization. To illustrate this, the motivational example hints at the exploitation of vulnerabilities in software used at both the power plant and the hospital. Information about the relevance can be very helpful for analysts when prioritizing CTI artifacts to be analyzed.

Contextual information about the user can either be collected by the platform host or can be found in STIX objects describing the user. Specific characteristics (e.g., the industry sector) of a CTI publisher and those of a consumer are assessed for matches. In addition, attribute values for available STIX objects—for example, the *Vulnerability*—can be compared with the user's characteristics (e.g., the applied technologies), too. The coverage ratio expressed in Eq. 9 indicates relevance by taking the sets of all property values for consumer PV_c , publisher PV_p and relevant STIX objects PV_o into consideration. Congruent property values are set in relation to the total number of property values available for comparison.

The metric for the DQ dimension of relevancy could be further extended by inclusion of information contained in STIX *Sighting* objects. These objects incorporate a number describing how many times the referenced object has been identified. Therefore, this fosters the assessment of relevancy as frequently seen objects (e.g., an *Malware* object) might indicate a high relevance of these objects. This assumption can be expressed in a weighting factor added to the general metric and thus improve DQ assessment.

$$RE(o) = \frac{|PV_c \cap (PV_p \cup PV_o)|}{|PV_p \cup PV_o|} \quad (9)$$

Schema completeness The general completeness of data is confined to the assessment of schema completeness in the context of CTI. To distinguish this data quality dimension from syntactic accuracy, we focus on optional attributes and their values as the STIX JSON schemes already allow to

assess the existence of required attributes. This aspect is covered by the DQ dimension of syntactic accuracy later on.

STIX-based threat intelligence can be assessed for schema completeness of individual optional attributes a_o . A missing optional attribute value $v(a_o)$ is identified and classified according to Eq. 10. A strict distinction between complete (i.e., with value) and incomplete (i.e., without value) attributes is enforced. Referring to the example in Sect. 3.2, the vulnerability could be described in more detail with an external reference to a specific Common Vulnerabilities and Exposures (CVE) entry. This optional information would help others to gain further information about the actual vulnerability, how it is exploited, and how it can be fixed. This would ultimately improve CTI quality significantly by making it easier for others to leverage the CTI. In a second step, schema completeness for an entire STIX object o builds upon the previously calculated completeness scores for included attributes. The ratio of filled optional attributes to the total number of optional attributes of an object represents the schema completeness metric as shown in Eq. 11.

$$SC(a_o) = \begin{cases} 1 & \text{if } v(a_o) \neq NULL \\ 0 & \text{else} \end{cases} \quad (10)$$

$$SC(o) = \frac{\sum_{a_o \in (o \cap A_o)} SC(a_o)}{|o \cap A_o|} \quad (11)$$

Syntactic accuracy The data quality dimension of accuracy contributes to the correctness of data. With focus on syntactic accuracy in the context of CTI, the data schema is of particular importance for quality assessment. Syntactic accuracy gives a first indication on the extend to which an object is aligned with its data format.

The OASIS consortium behind the STIX format provides a JSON schema for each object. This allows for an automated matching of objects against those schemes to assess syntactic accuracy. In general, this DQ dimension is measured based on the analysis of attribute values $v(a)$ with $a \in (A_r \cup A_o)$ being part of a domain D [16]. In application to STIX-based threat intelligence, we can use the existing JSON schemes and validate each attribute value against the schema definition. The domain D is derived from the JSON schema which provides data types and allowed values. The assessment for syntactic accuracy of each attribute value is expressed by Eq. 12. An overarching indicator for syntactic accuracy of an object o can, respectively, be calculated as shown in Eq. 13.

$$SA(a) = \begin{cases} 1 & \text{if } v(a) \in D \\ 0 & \text{else} \end{cases} \quad (12)$$

$$SA(o) = \frac{\sum_{a \in o} SA(a)}{|o \cap (A_r \cup A_o)|} \quad (13)$$

Timeliness In the context of CTI, time ascends to one of the crucial elements of CTI quality. As stated earlier, outdated intelligence is identified throughout the relevant literature as one of the core challenges [3,5,13]. It is quite evident that the most current and up-to-date CTI artifacts probably implicate the most value for any type of analysis.

Time-based information contained within CTI data builds the basis for the configuration of a timeliness metric applicable to the CTI context. In general, various metrics can be utilized to assess timeliness. Considering the STIX data format, a basic timeliness metric is described in Eq. 14. The two components of this metric—currency and volatility—are present in every STIX object or can be derived from inherent features of the CTI platform. Volatility in this setting is expressed by the number of modifications to the assessed STIX object. The number of modifications can be drawn if concepts like the historization from earlier work are implemented [29]. This concept allows to track changes and the number of changes applied to a STIX object. Currency is referring to the age of the information and thus the time since its last modification. However, this metric entails certain problems specifically with regard to interpretability as well as to other requirements [30].

Where statistical data about the decline of timeliness for specific CTI information does exist, the metric for timeliness must be adapted. Resulting values of a statistical timeliness metric shown in Eq. 15 can subsequently be interpreted as probability of up-to-date CTI information. Considering the example in Sect. 3.2, the decline for certain STIX objects is higher than for others. File hashes as in the *Indicator* of Listing 1 will likely have high decline values as, for example, malware binaries might undergo slight changes frequently leading to changed hash values. In contrast, information regarding the threat actor might not change in time, thus having no statistical decline at all.

In contrast to these metrics, specific assessment of STIX-based CTI for the DQ dimension of timeliness can also be based on characteristics of STIX objects. For example, *Sighting* objects can provide information about the time of occurrence of referenced STIX objects. It can be thus inferred that for the timeliness of referenced STIX objects, the concept of inheritance applies. STIX objects of type *Observed Data* can be assessed for timeliness following the same procedure. Our proposed metric described in Eq. 16 includes the current time, the time of last occurrence, and a predefined time-based threshold value to foster the applicability of timeliness to any given CTI use case. In general, we focus on objective metrics of timeliness. Subjective perceptions such as expert knowledge about threshold values assist the assessment and can be considered further during implementation. Referring back to the motivational example, the hospital's security analysts can define a threshold based on their experience that indicators are outdated after a specific amount of time.

$$TI_{Basic}(o) = \frac{1}{(Currency(o) \times Volatility(o)) + 1} \quad (14)$$

$$TI_{Statistical}(o) = \exp(-Decline(o) \times Currency(o)) \quad (15)$$

$$TI_{Assisted}(o) = \begin{cases} 1 & \text{if } t_{current} - t_{last} < threshold \\ 0 & \text{else} \end{cases} \quad (16)$$

5.2 Object level

On the object level, we consider two dimensions which rely on manual input and are therefore defined to be subjective dimensions. They center on object characteristics of a higher abstraction level and often follow a cross-object perspective.

Representational consistency In general, the assessment of representational consistency relies on a set of rules C and semantic conditions c_j contained therein for the underlying data [24]. This DQ dimension needs to be adjusted to the requirements of the individual context and the given use case. Analogous to schema completeness, representational consistency goes beyond aspects of syntactic accuracy. For the context of threat intelligence, representational consistency allows for the enforcement of additional formal requirements which are not addressed by the dimensions of syntactic accuracy or concise representation. These might originate from data format requirements or requirements imposed by a CTI sharing platform. In the following, we propose two exemplary conditions configured to the STIX data format. CTI platforms could define further conditions or adjust existing ones. This is part of an iterative approach to support an increasingly detailed assessment of representational consistency.

In the context of STIX-based threat intelligence, we suggest a first condition to represent the necessity of existence of referenced STIX objects. For all STIX objects, the following “inter-relation constraint” [16] applies: referenced objects of embedded relationships must exist. Moreover, considering individual STIX objects specific relationships must be verified. This applies for all SROs as they connect per definition two SDOs. A second exemplary condition takes time-based information and the chronological order of creation and modification of CTI into account. Hence, it must be verified on the “intra-relation constraint” level that the creation time of any object is prior or equal to the time of modification. Besides, SROs can connect two SDOs only after their creation. Creation time of the corresponding SDOs must be prior or equal to creation time of the SRO. Listing 1 reveals those two exemplary conditions for representational consistency, too. For the *Vulnerability*, modification time precedes creation time by a month. With regard to referenced objects, a *Relationship* (i.e.,

“relationship-3”) points toward a nonexistent *Vulnerability* (i.e., “vulnerability-2”).

The assessment of representational consistency on a condition basis is described in Eq. 17. A given STIX object is assessed for each defined condition $c_j \in C$ separately, and the results indicate if a condition is fulfilled. Representational consistency per object is aggregated over all defined conditions in the set of conditions C as seen in Eq. 18.

Please note that although the assessment of an object o regarding a condition c_j can be automated and therefore is objective, the definition of the respective conditions is fully in control of the responsible stakeholder. Thus, we interpret this dimension to rather be subjective than objective with respect to the definitions in Sect. 4.1.

$$c_j(o) = \begin{cases} 1 & \text{if } o \text{ fulfills condition } c_j \\ 0 & \text{else} \end{cases} \quad (17)$$

$$RC(o) = \prod_{j=1}^{|C|} c_j(o) \quad (18)$$

Reputation It is important to build trust in shared CTI environments. Trust and the assessment of trustworthiness can build upon the DQ dimensions of reputation, provenance, and believability. The introduction of two quality sub-dimensions for reputation—reputation of the publisher (i.e., provenance) and reputation of the data set (i.e., believability)—allows for a holistic coverage of the trustworthiness concept in the context of CTI exchange. Our proposed assessment is based on functionalities similar to reputation systems and external human input. Reputation scores for a given publisher p might adhere to a five-star rating system as shown in Eq. 19 as well as reputation scores of a STIX object o as shown in Eq. 20. Based on these reputation scores s contained in a set of scores S , an overall reputation $RS(x)$ for either publisher or STIX object is calculated according to a simple ratio function described in Eq. 21. Sample size $|S|$ supports data quality assessment further and constitutes a relevant additional data point. In the situational example, the hospital can articulate trust toward the power plant and its CTI by rating them accordingly.

While the above-mentioned configuration of reputation is purely subjective, possibilities exist to assist the quality assessment with objective metrics. For one, a list of trusted CTI publishers can be introduced as an indicator for the reputation of a publisher. An analogous indication for the reputation of an object is the number of access requests to a certain artifact set in relation to the number of CTI platform consumers having taken remediating steps upon the threat intelligence.

$$RS(p) = \{s \mid 1 \leq s \leq 5 \wedge s \in \mathbb{N}\} \quad (19)$$

$$RS(o) = \{s \mid 1 \leq s \leq 5 \wedge s \in \mathbb{N}\} \quad (20)$$

$$RS(x) = \frac{\sum_{s \in S} s_i}{|S|} \quad (21)$$

5.3 Report level

On the upmost level of Fig. 3, we place a single dimension which takes a complete STIX report including its contained SDOs and SROs into consideration.

Appropriate amount of data The requirement to include the appropriate amount of data quality dimension arose during our discussions with domain experts as described earlier. However, the application of a generic metric proves not feasible due to its semantic component in the form of needed data units. We therefore base our metric on the additional comments of security analysts. Homogeneous SDO types and very few relationships seemingly lead to the experts’ perception that the report in general is not very helpful.

To distinguish between a report with homogeneous STIX objects and one with rather diverse objects is a matter of implementation and cannot easily be compressed into a metric. As described above, this is a rather complex task which needs further research efforts. As a first approach toward a feasible support of security experts, we propose a clear representation of occurrences of each STIX object in an artifact. This is achieved by simply counting the instances of the different SDO types within a report. Visualization can provide this relevant information at the report level and can aid DQ assessment at first glance.

Besides this, we take graph theory for the connectedness of the STIX report’s SDOs into account. We argue that a metric based on the number of relationships can provide a basic indicator to assess this DQ dimension. In general, the metric for the DQ dimension of appropriate amount of data should yield a higher score for CTI which is densely connected. A given STIX report depicts a graph, and its contained SDOs represent vertices. SDOs are furthermore connected with each other through SROs which resemble edges from a graph perspective. The metric in Eq. 22 sets the number of existing SROs in a given STIX report in relation to the maximum possible number of SROs as defined by the number of SDOs for this report.

The metric for the appropriate amount of data is a challenge for future work. Our simplistic metric could be improved in different ways. A possible direction is a statistical comparison of all available reports. Calculating a report’s score for the diversity of SDOs and the respective relationships as a comparison with a baseline diversity from other reports might be a feasible direction. However, the prerequisite to this approach is a sufficiently high number of reports included into the baseline.

$$AD(r) = \frac{|sro \in r|}{\frac{|sdo \in r|(|sdo \in r| - 1)}{2}} \quad (22)$$

5.4 Aggregating quality indicators

The aggregation of DQ dimension scores for CTI has to be customizable as described earlier in Sect. 4.2. Adjustable aggregation parameters enable CTI consumers to define the weight of each of the DQ dimensions D in the procedure of calculating a quality indication for each STIX object. For this customizable aggregation, we propose a weighted average (see Eq. 23), where each dimensional score $d_i \in D$ is weighted with a parameter $w_i \in \mathbb{N}$. This parameter w_i can be adjusted by each platform consumer. If no custom value is provided for a dimension d_i , the default weight is $w_i = 1$.

To support consumers' decisions on which report available on a CTI sharing platform to analyze, we additionally propose a report quality indicator calculated following Eq. 24. This score contains the individual DQ object scores $DQ(o)$ and the additional report-level dimension of the appropriate amount of data $AD(r)$. Only the additional DQ dimension's score is weighted in this aggregation with $w \in \mathbb{N}$. The default value for w is again 1. Following this aggregation structure, the weight of each DQ dimension is adjustable by the platform users consuming the respective CTI to ensure that the quality scores represent their individual preference of the dimension's importance.

$$DQ(o) = \frac{\sum_{d \in D} d_i \cdot w_i}{\sum_{d \in D} w_i} \quad (23)$$

$$DQ(r) = \frac{(\sum_{o \in r} DQ(o)) + AD(r) \cdot w}{|r| + w} \quad (24)$$

6 Visualizing quality of CTI

Informing users of CTI about the quality of the intelligence at hand is of crucial importance. This is a vital task in the context of a sharing platform as it allows users to build trust toward the shared CTI. We argue that it is not enough to only inform users about the result of a CTI quality assessment. Instead, the assessment process itself must be transparent for security analysts. Thus, a visual interface should inform them "Why" a report has a specific quality score. As different aspects of CTI quality might also be of varying importance for users, the visual interface could also support parametrization of the quality aggregation as described earlier. Besides the need to inform users about the CTI quality and building trust, their subjective perception of a report's quality is highly relevant for the assessment process. Therefore, a solution is needed to allow them to share their opinion.

Providing a possible path to solve these requirements, we draw upon the idea to make complex threat intelligence exchange formats, like STIX, accessible for human experts through an interactive visual interface. The feasibility and applicability of this approach have been shown in earlier work [29]. In this work, we implemented and evaluated an open-source visual analytics prototype for STIX. We extend this proof of concept by including indicators about the CTI quality in the interface and by implementing functionalities for experts to share their subjective quality assessment where necessary. In the following sections, we briefly introduce the changes made to the original visual interface called Knowledge-Assisted Visual Analytics for STIX (KAVAS). Additionally, we extend the underlying database (CTI Vault) to integrate notions of threat intelligence quality. However, both the database and the visual representation are built to only handle data compliant to the STIX specification. Thus, before including CTI quality into the tool, a solution to represent CTI's quality in the STIX format is needed.

6.1 Integrating quality indicators into STIX

In its current specification, the STIX format has no object types or properties to model indications about the quality of CTI. However, the specification defines the format in a way which allows for the extension of the baseline specification [31]. This opens different possible ways to integrate CTI quality into this format. On the one hand, it is possible to define completely new types of STIX objects. On the other hand, additional properties could be added to the existing SDO and SRO types.

```
{
  ``type``: ``x-quality-indicator``,
  ``id``: ``x-quality-indicator--1``,
  ``created``: ``2019-07-25T09:00:00Z``,
  ``modified``: ``2019-07-25T09:00:00Z``,
  ``object_ref``: ``threat-actor--1``,
  ``measures``: [
    {
      ``dimension``: ``Syntactic Accuracy``,
      ``type``: ``objective``,
      ``score``: 0.8
    },
    ...
    {
      ``dimension``: ``Reputation``,
      ``type``: ``subjective``,
      ``score``: 0.7,
      ``rating_count``: 14
    }
  ]
}
```

Listing 2 Exemplary *Quality Indicator* object

Table 1 Definition of the *Measure* custom data type for STIX 2

Property name	Type	Description
dimension (<i>required</i>)	String	The dimension for which the measurement is described
type (<i>required</i>)	String (“ <i>subjective</i> ” or “ <i>objective</i> ”)	Describes whether the dimension’s score is based on a subjective or an objective metric
score (<i>required</i>)	Float	Double-precision number ranging from 0 to 1 describing the current result of the quality assessment for a quality dimension
rating_count (<i>optional</i>)	Integer	This property is only needed for “subjective” measures as it describes how many different ratings were given to produce the current score

Table 2 Definition of the *Quality Indicator* custom object for STIX 2

Common properties		
type, id, created_by_ref, created, modified, revoked, labels, external_references, object_marking_refs, granular_markings		
Quality indicator specific properties		
object_ref, measures		
Property Name	Type	Description
type (<i>required</i>)	string	The value of this property MUST be “x-quality-indicator”
object_ref (<i>required</i>)	identifier	Specifies the STIX Object that is referred to by this quality indicator
measures (<i>required</i>)	list of type measure	A list holding all measurements for the different quality dimensions available for the referred-to STIX Object

In any case for each STIX object, the calculated scores for the different quality dimensions need to be documented. To capture the necessary information in a STIX-conformant way, we therefore propose the custom data type *Measure* defined in Table 1. This data type consists of the name of a specific dimension and the object’s respective score. It is worth noting that our proposal centers on float values. Nevertheless, scores on an ordinal scale are also possible. Respective conversions can be implemented by defining ranges of float values which refer to a specific ordinal scale (low, medium, and high). Additionally, the custom data type contains the type (subjective or objective) of the dimension. For subjective dimensions, the count of received ratings used to calculate the score can be stored.

We opt to attach a list of measures structured according to the proposed *Measure* data type to a new Custom STIX object. While it is also possible to include this list in any existing STIX object, our proposal aims to maintain a clear separation between actual threat information and the related quality information. Additionally, this proposal produces as less interference as possible with the existing data model. Neither the existing SDOs nor SROs need to be changed. In compliance with the specification, we follow the mechanisms and requirements given to introduce custom objects called *Quality Indicator*. Besides the mandatory *Common Properties*, a number of specific properties are established [31].

Table 2 defines the proposed STIX Custom Object. We include common properties of our *Quality Indicator* object which are mandatory for each SDO. These properties are followed by several specific properties defined for the object. The last part of Table 2 defines allowed data types and values for the specific *Quality Indicator* properties. The *type* attribute must not hold other values than “x-quality-indicator”. The *Quality Indicator* object is not connected to any other objects with an explicit SRO but holds a property “object_ref” reflecting the ID of the SDO or SRO for which the object indicates the relevant quality measures. Finally, the object contains a list of “measures” which holds the scores for all the DQ dimensions. The list is formed of the custom *Measure* data type. An exemplary and simplified object is shown in Listing 2.

STIX is an actively maintained CTI standard. Recently, there have been developments that incorporate some aspects similar to our CTI quality concept within the newest STIX2.1 Committee Specification Draft.⁵ Most notably, this draft includes an *Opinion* SDO to capture perceptions by CTI consumers about the correctness of a STIX object. The *Opinion* SDO aims to document the level of agreement with the referred-to STIX object(s) on a Likert-type scale ranging from strongly disagree to strongly agree. As can be seen by the purpose and the description of the *Opinion* SDO, this spe-

⁵ <https://docs.oasis-open.org/cti/stix/v2.1/stix-v2.1.html>.

cific STIX object is another prospective option to implement elements of the *Reputation* data quality dimension. Nevertheless, in contrast to our proposed *Quality Indicator* SDO the draft and its *Opinion* SDO fall short to cover a larger CTI quality concept.

6.2 Persisting quality indicators in the CTI Vault

The original database for the CTI visualization is a graph-based approach based on Neo4J.⁶ This is quite reasonable as STIX is based in graph-like structure itself. Additionally, the integrity-preserving storage concept proposed by Böhm et al. [29] is most efficiently implemented using this technology. We extend this approach by adding a new database to the architecture. This new database is solely supposed to persist the *Quality Indicator* objects introduced in Sect. 6.1. As described, these objects do not have any explicit connections to other STIX objects via SROs. Their integration would double the number of objects inside the existing database and would certainly affect the performance negatively. Therefore, we decided to avoid storing the quality object inside the existing vault.

Our newly added “Quality Vault” is a document-oriented database (MongoDB⁷) for performance reasons. This additional vault persists the JSON representations of the *Quality Indicator* objects which are directly related to a single SDO or SRO in the CTI Vault via the “object_ref” attribute.

6.3 Displaying quality indicators in KAVAS

Throughout this section, we describe the changes we made to the original visual interface to include visual indications about the quality of STIX artifacts. In Böhm et al. [29], the process of visually analyzing STIX-based CTI with KAVAS starts with a simple drop-down menu to select the report of interest. The drop-down menu contains only the name of the report given by its publisher. This does not disclose any additional information to the analyst whether the report might be of interest or not. We changed this initial view of the KAVAS interface to be more informative and also to give first insight into the quality of the report. The visual interface now contains an expandable list of all available reports from the CTI Vault. The expansion panel for each STIX report consists of three main sections depicted in Fig. 4 informing analysts on the contents and overall quality of a STIX artifact at first glance⁸:

1. At first, a description (if given by the report’s producer) gives high-level information on what the report is about.
2. The second section shows which specific STIX objects, both SDOs and SROs, are contained in the report and how often they are present. Object types that are not present in the respective STIX artifact are grayed out. This view fulfills the requirement to provide a view on the homogeneity of a STIX artifact as described in Sect. 5.3 within the quality dimension of *Appropriate amount of data*.
3. The third section gives a very brief and high-level indication on the average quality of the STIX objects and their interconnectedness within the respective report using two gauge displays. This connectedness is represented by the score as described in Sect. 5.3.

After a STIX report is selected and its graph representation is loaded in the visual interface further changes become apparent, clicking a node or a link of the graph details its information in a details-on-demand card view. The original object card only contained a tab with the attribute values of the selected object and, for SDOs, a tab with its directly linked neighbors in the graph. We now add a quality badge in the header of the object card displaying the aggregated quality score of the dimensions from object and attribute level as described in Sect. 5.4. Furthermore, we add a new tab providing more detailed insight and transparency of the quality measuring. The new object quality tab on the details-on-demand view is shown in Fig. 5. Again, this component is divided into three sections:

1. A gauge visualization of the object’s overall quality score aggregated from the scores at the attribute and object level.
2. A section with progress bars indicating the object’s score for all described objective dimensions.
3. A third section that holds the indicators for an object’s scores of subjective quality dimensions. For this part of the quality tab, we need to both inform the user about the current score and allow them to provide their own subjective quality measurement for the respective dimensions. To do so, we lend from reputation systems and display a rating bar ranging from one to five stars which is a well-known visual metaphor in reputation systems. These rating bars always show the current overall score for the quality dimension (blue stars) in relation to the possible highest rating while also allowing users to click each of the stars to provide their own rating. Numbers in parentheses besides the name of the DQ dimension indicate the count of ratings provided by other users (e.g., the number of subjective assessments on which the current score is based).

⁶ <https://neo4j.com/>.

⁷ <https://www.mongodb.com/>.

⁸ Please note that the displayed information is computed based on a test data set which is different from the STIX example in Sect. 3.2.

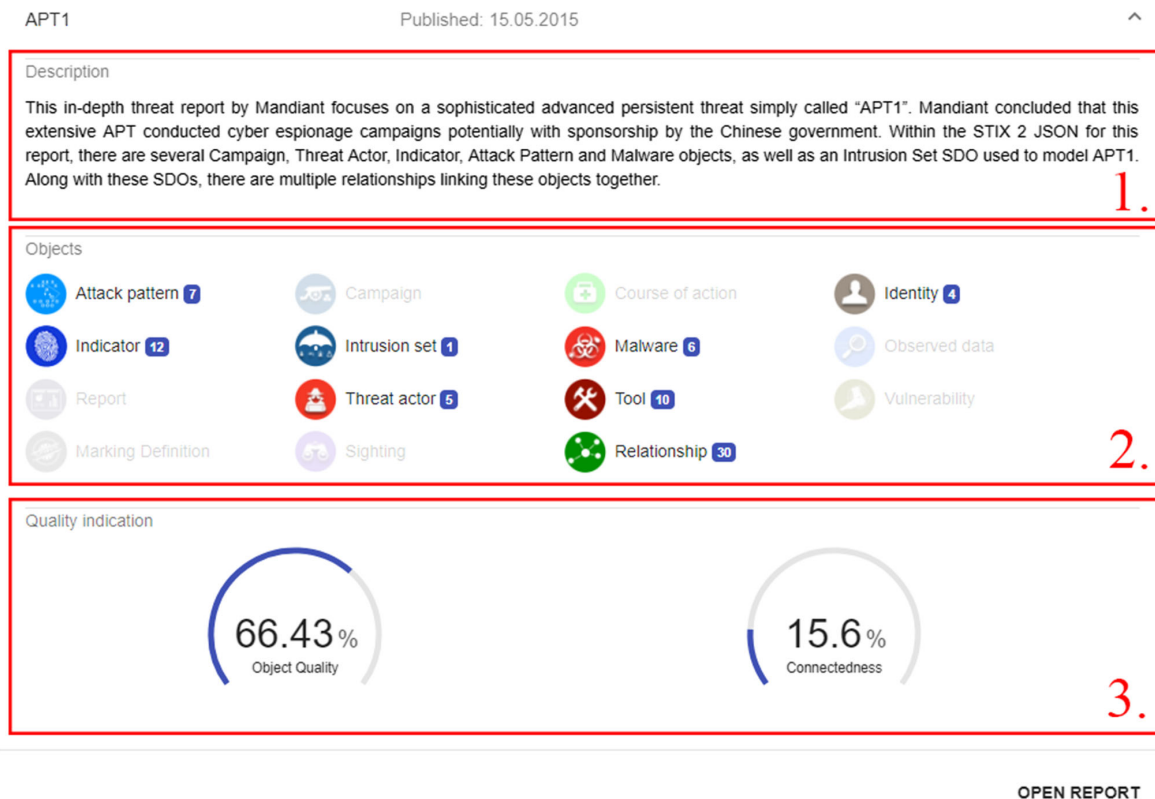


Fig. 4 View of report selection screen

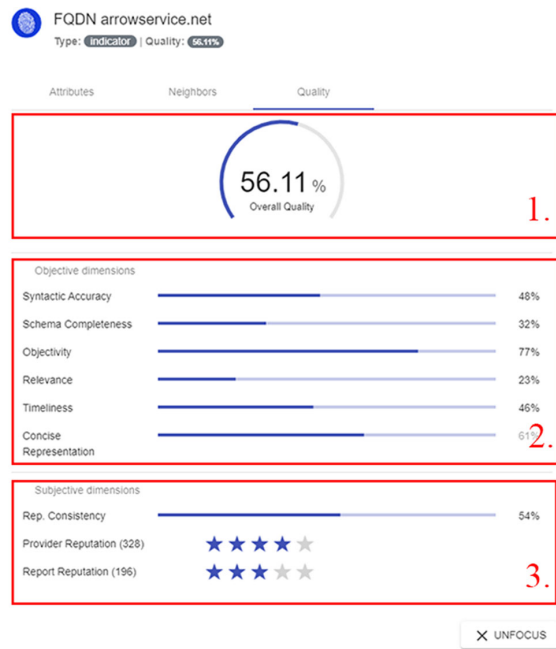


Fig. 5 Quality tab on object card (details-on-demand)

The quality tab fulfills a twofold goal: First, it makes the aggregation of the quality dimensions transparent, and second, it allows collecting user’s input for subjective quality dimensions. We actively decided not to use any color-coding for the scores. Traditionally, respective scores are colored with red (low quality), orange (medium quality), and green (high quality). However, we only aim to inform CTI analysts about the quality scores and do not want to provide any kind of interpretation of low or high score for any quality dimensions. As described earlier, this is mainly because the quality dimensions might be of different interest for different consumers. Therefore, low scores for respective dimensions of an object do not automatically implicate that the object is irrelevant or of low overall quality for the consumer.

In order to allow users to customize the aggregation of quality dimension scores following our previously described bottom-up approach, analysts need a way to define the dimensions’ weights. To provide this functionality, we extend the KAVAS settings dialog with a slider for each quality dimension as depicted in Fig. 6. The default configuration assumes that all dimensions are equally important (e.g., have a weight of 1). Analysts can use the sliders to customize the dimension aggregation according to their preference. If they do not want a specific dimension to have any influence in the aggreg-

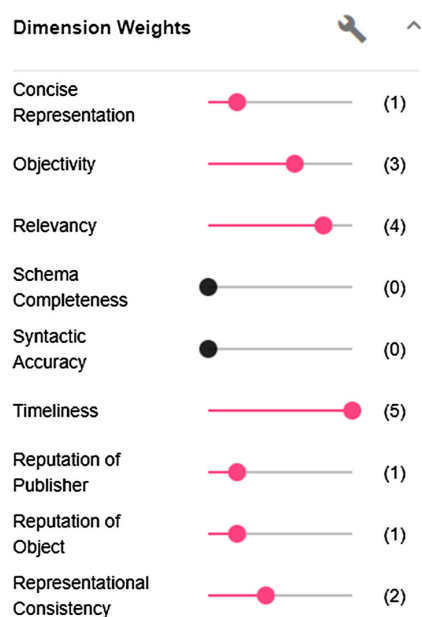


Fig. 6 Slider for dimension weights

gation, they can assign a weight of 0 and for a dimension with crucial importance, they can accordingly assign a weight of 5. Please note that the metric for dimension aggregation in Eq. 23 does not limit the range for the dimension's weight. However, we chose to limit them in the visual interface to a range from 0 to 5 for more practical feasibility.

6.4 Evaluating the visual display of CTI quality

To validate the visualization approach and to provide first evidence of its suitability, we conduct a number of expert interviews. The main goal of these interviews is to validate that the visual approach helps analysts to understand the DQ of the CTI artifact at hand.

Participants The interviewees are three security experts from different sectors and company sizes. We conduct interviews with two highly experienced security analysts from a big international conglomerate and a medium-sized manufacturing company. The third interviewee is a researcher focusing on CTI sharing formats. Each participant has a medium to high knowledge regarding threat intelligence as all of them deal with information security on a daily basis. None of the participants currently obtains a quality assessment on CTI.

Design and procedure The interviews with the experts are designed following a semi-structured approach and are split into the following four phases [32]:

1. *Introduction* Starting the interviews, each participant is questioned for some basic data, their experience, such as knowledge on CTI and DQ aspects. Afterward, each expert is introduced briefly to the STIX format (if necessary) and to the problem of measuring CTI quality. Thereby, the experts are actively asked to criticize any potential issues noticed throughout the following interview phases.
2. *Measuring CTI quality* In this phase, we aim to get additional feedback on the individual dimensions and the configured metrics for quality assessment of STIX artifacts (Sects. 4 and 5). Although the dimensions and the metrics are already the result of an iterative process where we collaborated with researchers and practitioners, an additional evaluation of these results is performed in this phase. The selected dimensions, their structure, and the configured metrics are discussed with the interviewees to identify whether they support the relevance of the proposed DQ measurement approach. We also ask the participants what aspects of the dimensions and metrics might need a more detailed explanation and whether they think that the metrics are comprehensible for security analysts without much prior knowledge in the DQ area.
3. *Visualizing CTI quality* The focus of this phase is to test the suitability of the proposed visualization approach. To enable the interviewees to work with the DQ visualization, we make use of sample STIX reports provided by the OASIS consortium. Prior to the interviews, these reports were manually fed into the existing KAVAS tool and enriched with the DQ measures. During the interviews, the participants can access the STIX reports through the extended KAVAS tool as described in Sect. 6. The main goal in this interview phase is to identify whether the proposed visualization elements to display the CTI quality are actually helpful for security analysts. We ask the interviewees whether the proposed DQ metrics are comprehensible with the chosen visualization elements and what further aspects they think would enhance the understanding of DQ assessment within CTI.
4. *Wrap-Up* The last phase of the interviews is dedicated to a summarizing discussion. Here, we discuss with the participants whether an implementation of the proposed metrics and the respective visualization approach would be applicable to operative deployment and the conditions thereto. Finally, we collect a list of ideas and features the interviewees find useful for improving our approach.

Results The interviews lasted between 45 to 75 minutes. The results of the conducted interviews are presented in the following, divided according to the four interview phases described before:

Table 3 General information on the interview participants

	Position	Business branch	Organization's size	CTI knowledge	DQ knowledge
#1	Senior security analyst	Manufacturing	ca. 400.000	High	Medium
#2	Head of security information management	Manufacturing	ca. 15.000	Low	Medium
#3	Security researcher	Academia	ca. 5.000	High	Medium

1. *Introduction* The results of the introduction phase are summarized in Table 3 giving an overview on general information about the interviewees.
2. *Measuring CTI quality* Above all, the interviewees unanimously stress the importance of metrics for quality within the field of CTI. Valuable and actionable CTI is stated to be highly dependent on quality and currently more often than not CTI is of low quality. A recurring theme mentioned in this phase by multiple interviewees is the interpretation of CTI quality. It is pointed out that the implementation of metrics for CTI quality by sharing platforms would benefit significantly from indication of low- and high- quality reference scores. Another identified theme is usability of DQ dimensions and metrics for CTI. Here, formally sound metrics, the chosen naming convention of DQ dimensions based on existing academic work and security analysts without DQ or mathematical background, stand opposite each other. Comprehensive explanations are seen as one approach to foster security analysts' understanding of the precise meaning of CTI quality dimensions and metrics.
3. *Visualizing CTI quality* All interviewees agree on the necessity to provide easy access to CTI quality through the use of visualization elements and validate our visualization approach. All interviewees agreed that the chosen visual representation allows for a quick recognition of CTI quality. They also uniformly considered the possibility to include subjective perceptions with means similar to reputation systems very helpful. Nevertheless, the interviewees name different extensions to the current visualization. For one, in-depth information about the DQ dimensions, the metrics, and possible interpretation is highlighted. Additionally, the showcased visualization includes percentages numbers and numeric weighting factors which could instead be visualized on a Likert-type scale. Another proposed extension targets the causal nature of low-quality scores. Visualization elements to detect improvements and eventually improve the CTI quality further are perceived as helpful. As one interviewee points out, user groups (e.g., system administrator or standard user) could be defined, given different permissions and thus see different visualizations.
4. *Wrap-up* In the final phase, the interviewees often come back to the timeliness dimension. The proposed metrics

for this DQ dimension needed additional explanations with regard to STIX specifics (i.e., Sighting SDO). Ideas and features mentioned by the interviewees to extend our work cover guidance to improve CTI quality and quality filtering with visualization elements. For instance, visual recommendations to reach a higher CTI quality (with or without prior knowledge about quality details) might be added to the current reactive assessment.

Overall, the interviewees' feedback indicates the valuable contribution of measuring and visualizing CTI quality. In particular, the dual approach itself (measure and visualize) is assumed to reduce complexity, lower quality assessment barriers, and foster CTI utilization. With regard to the implementation within a CTI sharing platform, we draw the conclusions that 1) there needs to be discussion on usability and adequate naming of DQ dimensions, 2) reference values are crucial for CTI quality interpretation, and 3) visual elements and textual explanations must be combined to avoid ambiguity.

7 Conclusion and future work

This work shed light on the assessment of DQ dimensions in the context of CTI. Nonetheless, there are further areas where research needs to be intensified and extended to.

7.1 Conclusion

Recent developments in the cyber threat landscape urge organizations to join forces against the adversaries. Collaboration based on the exchange of available threat intelligence arises as one of their most effective weapons. CTI sharing leveraged by respective platforms helps to spread knowledge about current threats. However, respective formats are oftentimes complex and large leading to a lack of readability for domain experts. Therefore, it is a vital task to help experts understand the CTI, for example, by providing visual representations. CTI can only be effective when security experts are able to comprehend it quickly and efficiently. Another issue hindering the effectiveness of CTI is the missing quality control on sharing platforms. This lack of DQ management mostly

stems from missing proposals to measure CTI quality in the first place.

Our studies cumulated within this work constitute a necessary first step into this direction. This includes the two focal points of measurement and visualization of threat intelligence quality. Existing academic work proposed sets of possibly relevant quality dimensions as well as high-level requirements for CTI quality assessment. Although calling for an inclusion of quality assessment and assurance into the world of CTI sharing, up to now there are no proposals for actual quality metrics applicable to CTI. Therefore, proposing a relevant set of quality dimensions and configuring respective metrics for a specific CTI format is a necessary step toward actionable CTI quality assessment. The proposed dimensions and metrics can help to build a cohesive quality management methodology for CTI based on the STIX data format. Most of our findings regarding suitable as well as not applicable DQ dimensions or metrics can also be applied to other CTI formats. It is possible to think of additional, more specific dimensions which could be defined to assess quality of threat intelligence. However, in this work we define a base set of dimensions that originate from existing and widely agreed-upon DQ dimensions. This base set can easily be extended, and detailed metrics can complement our proposed ones if necessary.

Besides the definition of metrics to measure CTI's quality for relevant dimensions, we also showed how this quality assessment can be made transparent to users of a sharing platform. Transparency herein supports both building trust for the available information and making informed decisions about which CTI artifact is worth analyzing. This is important as current sharing platforms already hold an unmanageable amount of threat intelligence. Informing potential consumers of an artifact about its quality is a helpful decision support for the consumer. The visual display of an object's overall quality including the respective scores for individual quality dimensions helps consumers to understand how the DQ measurement result was reached. Additionally, it provides a way to collect important input from users for subjective quality dimensions. We therefore also show how human CTI analysts can be included into the quality assessment.

7.2 Future work

Our work can be seen as a first step into the direction of measuring CTI quality. However, we can identify several topics demanding additional research effort.

We are among the first to propose a cohesive set of applicable CTI quality dimensions. Therefore, these dimensions might be subjected to changes as more knowledge is gained about CTI sharing processes, platforms, and associated stakeholders. One dimension which needs further attention is the *Appropriate amount of data*. The proposed metric is a first

approach toward a highly complex issue. It is difficult to define which amount of data—either data regarding STIX objects or the information described by these objects—is appropriate. Thus, we propose a simple metric to give domain experts an indication of the data contained in a STIX report. The DQ metric for the appropriate amount of data should be further detailed upon analysis and verification with CTI platform data. Furthermore, the metrics to evaluate quality should be reconfigured for other CTI formats and integrated into a cohesive data quality management methodology for CTI.

After formally configuring the metrics for the selected quality dimensions, those metrics should be implemented into an actual sharing platform. Up to now, we only tested them in a small scaled environment. A complete implementation will likely raise further issues about the selection of suitable algorithms and the control of user participation and intentions which go beyond the core DQ assessment and have not been addressed in this work. Warranted through an implementation, the extension of some proposed dimensions can become feasible as more information about the requirements will be available. Implementing and extending the dimensions and metrics are necessary steps to finally build a cohesive methodology for quality assessment of CTI including processes to assure and improve quality of artifacts on a sharing platform.

Acknowledgements Open Access funding provided by Projekt DEAL.

Funding Part of this research was supported by the Federal Ministry of Education and Research, Germany, as part of the BMBF DINGfest project (<https://dingfest.ur.de>). Part of this project has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 830927.

Compliance with ethical standards

Conflict of interest All authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Symantec Corporation.: Internet security threat report 2019 (2019). <https://www.symantec.com/content/dam/symantec/docs/reports/istr-24-2019-en.pdf>
2. Riesco, R., Villagrà, V.A.: Leveraging cyber threat intelligence for a dynamic risk framework. *Int. J. Inf. Secur.* **18**, 715–739 (2019)
3. Ponemon Institute LLC.: Live threat intelligence impact report 2013 (2013). <https://www.ponemon.org/blog/live-threat-intelligence-impact-report-2013-1>
4. Ring, T.: Threat intelligence: Why people don't share. *Comput. Fraud Secur.* **2014**(3), 5 (2014)
5. Sillaber, C., Sauerwein, C., Mussmann, A., Breu, R.: Data quality challenges and future research directions in threat intelligence sharing practice. In: Proceedings of the 2016 ACM on Workshop on Information Sharing and Collaborative Security - WISCS'16, pp. 65–70. ACM, New York (2016)
6. Sillaber, C., Sauerwein, C., Mussmann, A., Breu, R.: Towards a maturity model for inter-organizational cyber threat intelligence sharing: A case study of stakeholder's expectations and willingness to share. In: Proceedings of Multikonferenz Wirtschaftsinformatik (MKWI 2018), pp. 6–9. Springer, Heidelberg (2018)
7. Tounsi, W., Rais, H.: A survey on technical threat intelligence in the age of sophisticated cyber attacks. *Comput. Secur.* **72**, 212–233 (2018)
8. Juran, J.M., Gryna, F.M.: Juran's Quality Control Handbook, 4th edn. McGraw-Hill, New York (1988)
9. Jøssang, A., Ismail, R., Boyd, C.: A survey of trust and reputation systems for online service provision. *Decis. Support Syst.* **43**(2), 618 (2007)
10. Dandurand, L., Serrano, O.S.: Towards improved cyber security information sharing. In: 2013 5th International Conference on Cyber Conflict (CYCON 2013). IEEE Computer Society Press, Los Alamitos (2013)
11. Serrano, O., Dandurand, L., Brown, S.: On the design of a cyber security data sharing system. In: Proceedings of the 2014 ACM Workshop on Information Sharing & Collaborative Security - WISCS '14, pp. 61–69. ACM, New York (2014)
12. Kokulu, F.B., Soneji, A., Bao, T., Shoshitaishvili, Y., Zhao, Z., Doupé, A., Ahn G.J.: Matched and mismatched socs: a qualitative study on security operations center issues. In: Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (Association for Computing Machinery, New York, NY, USA, 2019), CCS '19, pp. 1955–1970. <https://doi.org/10.1145/3319535.3354239>
13. Sauerwein, C., Sillaber, C., Mussmann, A., Breu, R.: Threat intelligence sharing platforms: an exploratory study of software vendors and research perspectives. In: Proceedings of the 13th International Conference on Wirtschaftsinformatik, pp. 837–851. Springer, Heidelberg (2017)
14. Menges, F., Pernul, G.: A comparative analysis of incident reporting formats. *Comput. Secur.* **73**, 87–101 (2018)
15. Piazza, R., Wunder, J., Jordan, B.: StixTM version 2.0. part 2: Stix objects (2017). <https://docs.oasis-open.org/cti/stix/v2.0/stix-v2.0-part2-stix-objects.html>
16. Batini, C., Cappiello, C., Francalanci, C., Maurino, A.: Methodologies for data quality assessment and improvement. *ACM Comput. Surv.* **41**(3), 1 (2009)
17. Skopik, F., Settanni, G., Fiedler, R.: A problem shared is a problem halved: a survey on the dimensions of collective cyber defense through security information sharing. *Computers & Security* **60**, 154–176 (2016)
18. Batini, C., Scannapieco, M.: Data and Information Quality: Dimensions, Principles and Techniques. Springer, Cham (2016)
19. Wand, Y., Wang, R.Y.: Anchoring data quality dimensions in ontological foundations. *Commun. ACM* **39**(11), 86 (1996)
20. Wang, R.Y., Strong, D.M.: Beyond accuracy: What data quality means to data consumers. *J. Manag. Inf. Syst.* **12**(4), 5 (1996)
21. Redman, T.C.: Data Quality for the Information Age. Artech House Publishers, Norwood (1996)
22. Umbrich, J., Neumaier, S., Polleres, A.: Quality assessment and evolution of open data portals. In: 2015 3rd International Conference on Future Internet of Things and Cloud (FiCloud), pp. 404–411. IEEE Computer Society Press, Los Alamitos (2015)
23. Wang, R.Y., Storey, V.C., Firth, C.P.: A framework for analysis of data quality research. *IEEE Trans. Knowl. Data Eng.* **7**(4), 623 (1995)
24. Pipino, L.L., Lee, Y.W., Wang, R.Y.: Data quality assessment. *Commun. ACM* **45**(4), 211 (2002)
25. Batini, C., Palmonari, M., Viscusi, G.: The many faces of information and their impact on information quality. In: Proceedings of the 17th International Conference in Information Quality (ICIQ 2012), pp. 212–228. MIT, Cambridge (2012)
26. Sänger, J., Richthammer, C., Pernul, G.: Reusable components for online reputation systems. *J. Trust Manag.* **2**(5), 1 (2015)
27. Gascon, H., Grobauer, B., Schreck, T., Rist, L., Arp, D., Rieck, K.: Mining attributed graphs for threat intelligence. In: Proceedings of the 7th ACM on Conference on Data and Application Security and Privacy, pp. 15–22. ACM, New York (2017)
28. Chaturvedi, I., Cambria, E., Welsch, R.E., Herrera, F.: Distinguishing between facts and opinions for sentiment analysis: survey and challenges. *Inf. Fus.* **44**, 65 (2018)
29. Böhm, F., Menges, F., Pernul, G.: Graph-based visual analytics for cyber threat intelligence. *Cybersecurity (Cybersecurity)* **1**, 1 (2018)
30. Heinrich, B., Kaiser, M., Klier, M.: How to measure data quality? A metric-based approach. In: ICIS 2007 Proceedings pp. 108–122 (2007)
31. Piazza, R., Wunder, J., Jordan, B.: StixTM version 2.0. part 1: Stix core concepts (2017). <https://docs.oasis-open.org/cti/stix/v2.0/stix-v2.0-part1-stix-core.html>
32. Lazar, J., Feng, J.H., Hochheiser, H.: Research Methods in Human-Computer Interaction. Morgan Kaufmann, Burlington (2010)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

3 CTI-SOC2M2 – The quest for mature, intelligence-driven security operations and incident response capabilities

Publication information

Current status: Published

Journal: Computers & Security

Date of acceptance: 16 September 2021

Full citation: SCHLETTE, D., VIELBERTH, M., & PERNUL, G. (2021). CTI-SOC2M2–The quest for mature, intelligence-driven security operations and incident response capabilities. *Computers & Security*, 111, 102482, pp. 1-20.

Authors' contributions:	Daniel Schlette	45%
	Manfred Vielberth	45%
	Günther Pernul	10%

Journal description: Computers & Security is the most respected technical journal in the IT security field. With its high-profile editorial board and informative regular features and columns, the journal is essential reading for IT security professionals around the world.

Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/coseComputers
&
Security

TC 11 Briefing Papers



CTI-SOC2M2 – The quest for mature, intelligence-driven security operations and incident response capabilities



Daniel Schlette^{1,*}, Manfred Vielberth, Günther Pernul

University of Regensburg, Universitätsstraße 31, Regensburg 93053, Germany

ARTICLE INFO

Article history:

Received 30 April 2021

Revised 7 September 2021

Accepted 15 September 2021

Available online 21 September 2021

Keywords:

Maturity model

Cyber threat intelligence (CTI)

Security operations center (SOC)

Incident response

Security orchestration

Automation and response (SOAR)

Cybersecurity

ABSTRACT

Threats, cyber attacks, and security incidents pertain to organizations of all types. Everyday information security is essentially defined by the maturity of security operations and incident response capabilities. However, focusing on internal information only has proven insufficient in an ever-changing threat landscape. Cyber threat intelligence (CTI) and its sharing are deemed necessary to cope with advanced threats and strongly influence security capabilities. Therefore, in this work, we develop CTI-SOC2M2, a capability maturity model that uses the degree of CTI integration as a proxy for SOC service maturity. In the course, we examine existing maturity models in the domains of Security Operations Centers (SOCs), incident response, and CTI. In search of adequate maturity assessment, we show threat intelligence dependencies through applicable data formats. As the systematic development of maturity models demands, our mixed methodology approach contributes a new in-depth analysis of intelligence-driven security operations. The resulting CTI-SOC2M2 model contains CTI formats, SOC services and is complemented with an evaluation through expert interviews. A prototypical, tool-based implementation is aimed to document steps towards the model's practical application.

© 2021 Elsevier Ltd. All rights reserved.

1. Introduction

Despite the proliferation of advanced systems for cyber defense, it remains a major challenge to build, assess and improve security operations and incident response capabilities within an organization (Ahmad et al., 2021). This situation is paired with sophisticated threat actors in constant search for unprepared and insecure organizations. Besides, a cybercrime economy is monetizing victims' information and vulnerabili-

ties. Consequently, organizations are forced to implement adequate security operations. As different attackers exchange information about publicly known vulnerabilities, exploits, and successful tactics, this is a call to action for information security defenders.

In coping with attackers, the sharing of Cyber Threat Intelligence (CTI) has emerged as an essential measure (Brown et al., 2015). The benefits of collaboration and sharing of contextualized security information about threats, cyber attacks, and security incidents are additional external insights

* Corresponding author.

E-mail addresses: daniel.schlette@ur.de (D. Schlette), manfred.vielberth@ur.de (M. Vielberth), guenther.pernul@ur.de (G. Pernul).

¹ www.go.ur.de/ifs.

<https://doi.org/10.1016/j.cose.2021.102482>

0167-4048/© 2021 Elsevier Ltd. All rights reserved.

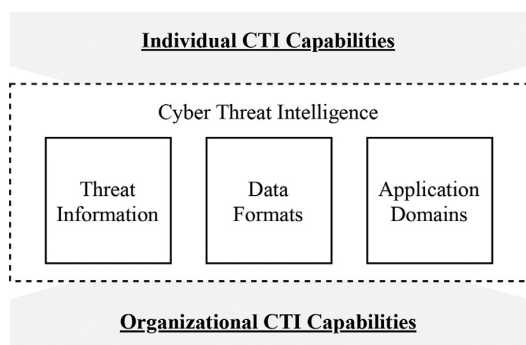


Fig. 1 – Cyber Threat Intelligence concept and capabilities.

and faster, intelligence-driven response. In essence, one organization's security incident description is another organization's threat intelligence. However, using CTI demands a common representation realized with structured data formats and frameworks. Therefore, to leverage CTI formats' full potential, these formats must be sufficiently integrated into organizational processes and tools. CTI is defined as being both actionable threat information (i.e., CTI artifacts) and a comprehensive concept that relates to individuals (i.e., security experts) (Shin and Lowry, 2020) and organizations with security services (see Fig. 1 and Section 3.2).

Security Operations Centers (SOCs) constitute a crucial element bridging CTI and organizational integration (Zimmerman, 2014). By definition, a SOC bundles organizational security roles, essential security services, and tools. Thus, in general, a SOC is responsible for CTI but sometimes dedicated organizational CTI units exist (Brown and Lee, 2021). CTI and its formats enable a well-functioning and effective SOC. While interacting primarily with SOC services, CTI formats also build the foundation of various technologies, such as Security Information and Event Management (SIEM) systems, Intrusion Detection and Prevention Systems (IDS/IPS), or Threat Intelligence Sharing Platforms (TISP).

In order to avoid the proverbial search for the needle in the haystack, SOCs depend on (external) support. We argue that this is possible primarily through a more data-driven approach to detecting and responding to security incidents. Therefore, mature security operations and incident response capabilities in modern organizations rely on data sources integrated via CTI formats (Kokulu et al., 2019). Current studies indicate a need for systematic integration with organizations, technology, and individuals (Lakshmi et al., 2021).

To improve an existing SOC, assessing its current state and finding deficiencies are pivotal objectives. These objectives can be achieved using a Capability Maturity Model (CMM), which fits the determined purpose and scope. Typically, CMMs combine a defined, rigorous academic methodology for development and a practical use case for which the model aims to assess and improve organizational capabilities (de Bruin et al., 2005).

In this work, we seek to better integrate Cyber Threat Intelligence and Security Operations Center. The challenge of assessing and improving intelligence-driven SOC maturity is addressed with an integrated approach. We thereby target stakeholders such as SOC Managers and Chief Information Security Officers (CISO) responsible for an organization's information security management and tactical security operation. With our proposed capability maturity model CTI-SOC2M2 we contribute a comprehensive method towards CTI-based SOC maturity focused on SOC services.

The iterative development methodology we apply leads to the following elementary steps:

- We analyze existing maturity models in the domains of SOC, CTI, and incident response.
- We develop SOC services based on a literature corpus.
- We map CTI formats and SOC services and build an integrated capability maturity model.

The remainder of this paper is structured as follows. Section 2 shows a motivating example and outlines the need for mature SOC services. In Section 3 we introduce SOC and CTI details. Capability maturity model development, including the methodology of this paper, is presented in Section 4. Then, Section 5 covers related maturity models and introduces our integrated three-tiered CTI-SOC2M2 architecture based on CTI formats mapped to SOC services. Maturity assessment with the proposed model and a prototypical implementation of the self-assessment tool build Section 6. Evaluating relevance and applicability form Section 7. Section 8 discusses contributions and limitations, while Section 9 concludes this paper.

2. Motivating example

In early 2021 unknown vulnerabilities in Microsoft Exchange Servers were detected and exploited by various threat actors. In the following section, we use this real-world attack to illustrate the necessity of adequate SOC services and the effective use of CTI and CTI formats. The example emphasizes beneficial aspects of threat intelligence for security operations and the difference between mature and immature SOC services.

Investigations of the Microsoft Exchange Server hack revealed both the timeline of events (Krebs, 2021) and the elementary steps of the attack (Microsoft Threat Intelligence Center (MSTIC), 2021). Based on the detection of 4 vulnerabilities and at first absent and later delayed patching by affected organizations, threat actors (e.g., Hafnium) performed the following actions:

1. **Scan** – The attacker first performs network scans for on-premises Microsoft Exchange Servers with versions susceptible to vulnerability CVE-2021-26855.
2. **Authentication bypass** – The attacker then uses server-side request forgery (SSRF) to bypass authentication and access the server.
3. **Remote Code Execution** – The attacker then uses additional vulnerabilities (e.g., CVE-2021-26857 and CVE-2021-26858) to run code and write files.

4. **Post-exploitation** – The attacker finally installs web shells used for command-and-control communication, exfiltrates information, escalates privileges, drops ransomware, and performs lateral movement.

The compromise of Microsoft Exchange Servers became possible due to threat actors exchanging information about vulnerabilities and exploits before SOCs had access to information on detecting and mitigating the attack. Besides, a lack of vulnerability management and security monitoring supported the widespread exploitation of organizations worldwide.

As it is recommended to patch information systems as fast as possible, situations such as the Microsoft Exchange Server hack point to additional mandatory security operations and the use of CTI and CTI formats. CTI provides the means to detect and mitigate any security compromise swiftly. CVE-IDs have been published on March 2nd 2021² and can be used by SOCs to be aware of the attacker's scan and authentication bypass actions. However, numerous organizations still were breached due to missing processes incorporating CTI. On AlienVault's Open Threat Exchange, CVE-IDs and various indicators of related adversary activity were quickly gathered³. This aggregated CTI can be retrieved using CTI formats such as OpenIOC 1.1 or STIX2.1. Likewise, other CTI sharing platforms list CTI on precise exploits and support export with TAXII⁴. Besides, courses of action to mitigate a compromised Microsoft Exchange Server and connected organizational networks are helpful to pursue incident response⁵. Therefore, it is in the best interest of any SOC to incorporate external CTI and have a thorough understanding of its formats.

Consequently, mature SOC services driven by threat intelligence allow organizations to better defend and mitigate post-exploitation actions. In contrast, SOCs missing external CTI face the complex task of detecting abnormal behavior solely from logs and events. As numerous breaches show, various organizations still struggled to cope with the situation long after CTI could be used. Again, this fact documents the need for mature SOC services.

3. Background

A thorough understanding of information security operations requires consideration of associated organizational concepts. The state-of-the-art of Security Operations Centers (SOCs) and incident response are detailed in the following. Next to these organizational aspects and processes, Cyber Threat Intelligence (CTI) is described. The foundations of threat intelligence are data, data formats, and data sources and relate to security operations.

² <https://nvd.nist.gov/vuln/detail/CVE-2021-26855>.

³ <https://otx.alienvault.com/pulse/6079bf21c21b824801b7a2a5>.

⁴ <https://exchange.xforce.ibmcloud.com/collection/In-the-Wild-Exploits-Seen-Targeting-MS-Exchange-8ec52986bb85fd000a3cf396677fbc1c>.

⁵ <https://github.com/microsoft/CSS-Exchange/tree/main/Security>.

3.1. Security operations center and incident response

To protect IT assets, today's organizations use SOCs. Vielberth et al. (2020) define a SOC as an organizational aspect consisting of four building blocks: people, processes, technology, and governance and compliance. Thereby, governance and compliance provide an encompassing framework. The primary goal of a SOC is to manage and enhance an organization's overall security posture, which usually cannot be achieved by a single entity or system. Instead, it requires a more complex structure. One cause of complexity is the plethora of SOC activities. A SOC creates situational awareness, mitigates security-related risks, and helps to fulfill regulatory requirements. Besides, SOC roles such as SOC analyst or SOC manager are required. The SOC roles are connected to the use of appropriate tools. For instance, SOC analysts typically use SIEM systems to analyze events and identify potential security incidents (Onwubiko, 2015; Zimmerman, 2014).

Of particular interest for both researchers and practitioners is incident response and its relation to SOC. It remains an open question whether incident response is part of a SOC. Two points of view are represented in literature with justifying arguments.

First, if one considers the 24/7 nature of a SOC, security analysts work in shifts around the clock. These analysts mainly analyze events in order to identify possible security-relevant events. Here, incident response and the actions performed are not part of a SOC as they follow incident detection. It can be argued that a dedicated Computer Security Incident Response Team (CSIRT) becomes active only in the case of an incident. Thus, in practice, CSIRT employees usually pursue main activities differently than responding to incidents (Ahmad et al., 2012). In contrast, SOC analysts pursue the analysis of security events full-time, depending on the company's size. This separation between SOC and incident response is based on two components: time and roles.

Second, if one considers the capabilities and activities combined within a SOC, there is an overlap between SOC and incident response, and the clear distinction becomes difficult. Primary SOC tasks such as detection and analysis of incidents and targeted incident response depend upon each other and include feedback loops. Thus, it is necessary to strive for a strong interconnection, if not integration, of SOC and CSIRT. To achieve a strong integration of those two sub-areas, it is of central importance to exchange and manage relevant threat intelligence effectively and efficiently (Onwubiko and Ouazane, 2020).

Finally, the inter-connectedness of SOC and incident response is documented within the incident response life cycle. In essence, this common concept to describe incident response includes several steps iterated in the process (Ab Rahman and Choo, 2015). In the case of the Incident Response Life Cycle developed by the National Institute of Standards and Technology (NIST) (Cichonski et al., 2012) there are four elementary steps. Based on Preparation, incident Detection & Analysis are conducted. These steps are followed by Containment, Eradication & Recovery. Then, the last step covers Post-Incident Activity. For all steps, feedback to previous steps is envisioned. Due to the elements outlined above, we de-

side to take on an integrated SOC and incident response perspective.

3.2. Cyber threat intelligence

Security information and threat reports build the basis of Cyber Threat Intelligence (CTI). The various types of threat intelligence range from *Indicators of Compromise (IoCs)* to *Tactics, Techniques and Procedures (TTPs)* and mitigating *Courses of Action (CoAs)* (Mavroeidis and Bromander, 2017). In addition, security information concerning vulnerabilities, exploit targets, risks and attack attribution are also considered CTI. While the most prominent examples of CTI are the more technical IoCs such as malicious IP addresses, domain names, and malware hashes, CTI includes the means to describe essentially any actionable information related to cyber attacks and security incidents (Tounsi and Rais, 2018).

Besides the threat information itself, the concept of CTI refers to processes to derive relevant knowledge from observed data. Initially, CTI is generated through detailed analysis and contextualization. Then, CTI sharing, including collaboration and dedicated platforms, builds another elementary part of CTI (Skopik et al., 2016). Ultimately, CTI is aimed to contribute to cyber defense by fostering security assessment and improving defensive security measures. Its use, therefore, targets decision-making processes and security operations.

CTI sharing is an essential application domain within the CTI concept and connects it with SOC. CTI sharing involves at least two parties: A producer and a consumer. While a CTI producer (e.g., a security analyst of a manufacturing company or a security vendor) creates CTI based on evidence, a CTI consumer retrieves external threat intelligence for further use. Besides informal and bilateral CTI sharing, dedicated platforms and communities can be involved. For example, an Information Sharing and Analysis Center (ISAC) can collect and distribute CTI to its member organizations via a Threat Intelligence Sharing Platform (TISP). In the context of CTI sharing, employees of SOCs produce and consume CTI on behalf of their organization.

CTI sharing and other application domains demand standardization. On a more granular level, data formats ensure standardization and support certain aspects of data quality assessment (Schlette et al., 2021). They are used to structure the different types of security-relevant information and the threat reports themselves. The role of different CTI formats has been explored by research highlighting their importance as a driving force for specific CTI use cases (Dandurand et al., 2014; Menges and Pernul, 2018).

Finally, associated with the CTI concept, two levels of capabilities can be identified – individual CTI capabilities and organizational CTI capabilities (see Fig. 1). Shin and Lowry (2020) capture the individual-level CTI capabilities and define three distinct CTI capability dimensions required for CTI practitioners to handle CTI artifacts effectively. Based on the Triarchic Theory of Intelligence (TTI), analytical or component intelligence, practical or contextual intelligence, and experiential intelligence represent practitioners' skills of the CTI capability comprising them. In contrast, we propose a capability maturity model that is centered on organizational-level capabilities. Our model has a different, more technical

Table 1 – Two perspectives on progress: the capability and maturity levels (adopted from the CMMI Product Team (2010)).

Level	Capability	Maturity
Level 0	Incomplete	
Level 1	Performed	Initial
Level 2	Managed	Managed
Level 3	Defined	Defined
Level 4		Qualitatively Managed
Level 5		Optimizing

focus integrating CTI and SOC. Nevertheless, its CTI formats, SOC services, and application domains relate to the analytical capabilities of the individual and thus link the two CTI capability levels.

4. Capability maturity model development

Although different maturity models exist, the primary objectives remain the same. As such, a maturity model serves four purposes (Ahern et al., 2004; Becker et al., 2009; de Bruin et al., 2005; CMMI Product Team, 2010): it provides a framework for assessing the current state of an object of interest in terms of its capabilities and other indicators of maturity; it allows the measurement of progress along a path of maturity stages, thus indicating a way for improvement; lastly, it allows benchmarking, though this requires an elaborated maturity model as well as widespread use. Since a SOC can generally be understood as a service provider (Zimmerman, 2014), a closer look at CMMI-SVC's structure and content is warranted (CMMI Product Team, 2010). The model framework consists of various process areas (PAs) or services divided into several thematic service categories. A *service* is defined by its purpose, specific and generic goals, as well as specific and generic practices to achieve those goals. Broadly speaking, whereas specific goals and practices denote the particular features of a service, the generic goals and practices apply across all services and denote the service's level of institutionalization. Institutionalization, in turn, is understood as the level of integration of the process in the overall organization and its consistent performance.

For assessing the current state, the CMMI provides two perspectives: whereas the first provides a view on the capability level of individual services, the second provides a view of the maturity level of multiple services across the organization. As such, an individual service's capability can progress from incomplete to performed, managed, and finally to defined - corresponding to levels 0–3, respectively (cf. Table 1). On the other hand, the organization's maturity can progress from initial to managed, defined, quantitatively managed, and finally to optimizing - corresponding to levels 1–5, respectively.

4.1. Methodology

The proliferation and development of maturity models have considerably increased since the inception of the most popular maturity model: the Capability Maturity Model (CMM) go-

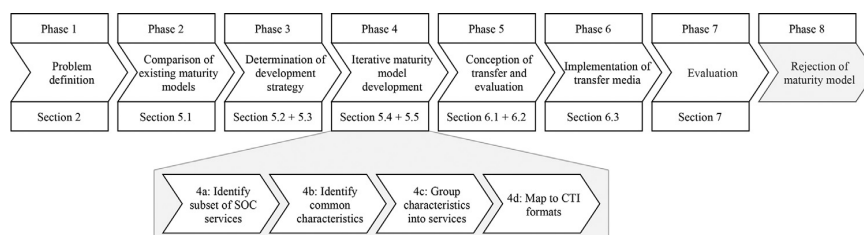


Fig. 2 – Methodology.

ing back to [Humphrey \(1988\)](#). However, as early as 2005, research has pointed out the lack of methodological rigor in maturity model development ([Becker et al., 2009](#); [de Bruin et al., 2005](#); [Mettler, 2009](#)). As [Mettler \(2009\)](#) points out, this is not just a concern for academics. Like [de Bruin et al. \(2005\)](#) before, he argues that the absence of a theoretical basis not just limits the generalizability but also strongly questions the validity of actual appraisals undertaken on the basis of such maturity models. Consequently, practitioners may face results that fail to deliver the promised benefits: acquiring a comprehensive picture of the current state of the object of interest and a path forward detailing where progress is needed to achieve higher stages of maturity.

We address the concerns raised by existing literature and use the methodology shown in [Fig. 2](#) for developing our integrated CTI-SOC2M2. Also, the structure of our paper follows the methodology. It is largely based on the widely-used methodology for maturity model development proposed by [Becker et al. \(2009\)](#). However, since the CMM methodology does not deal with the delimitation of services - which is an essential contribution of CTI-SOC2M2 - in detail, we have supplemented it with [Nickerson et al. \(2013\)](#)'s methodology for taxonomy development. The resulting eight phases of the methodology are briefly described in the following:

Phase 1, problem definition, prescribes that the target domain and the audience of the maturity model be made explicit. In addition, the problem relevance must be justified. In other words, the scope should be clearly defined ([de Bruin et al., 2005](#)). All of these requirements are met in the introductory section and emphasized with the motivating example.

Phase 2, comparison of existing maturity models, serves two purposes. First, it makes sure that research efforts are not wasted on problems that have already been solved. Second, it provides a sound basis for the following phase of determining the development strategy.

Phase 3, determination of development strategy, is entered once it is clear that no existing model suffices. Consequently, [Becker et al. \(2009\)](#) discern between four general development strategies: 1. the design of a completely new model; 2. the enhancement of existing models; 3. the combination of several models into a new one; and finally, 4. the transfer of structures or content from existing models to new domains. We find the principle of cumulative knowledge development applicable. Therefore, there is no need to design a maturity model from scratch, as proposed by strategy one. Further, plenty of literature has used the basic structure and generic terminology pro-

vided in prominent maturity models like CMMI ([CMMI Product Team, 2010](#)). The most sensible strategy for the present case is deemed to be strategy four.

Phase 4, iterative maturity model development, is divided into four steps: selecting the design level, selecting the approach, designing the level, and finally testing the result ([Becker et al., 2009](#)). As an approach to delimiting the SOC services, we have followed the methodology of [Nickerson et al. \(2013\)](#). Thereby the "empirical-to-conceptual approach" applies because of a well-established body of knowledge in the area of SOC. Then, to grasp this knowledge, a structured literature review is conducted. The adapted methodology of [Nickerson et al. \(2013\)](#) comprises the three steps to identify the subset of SOC services, identify common characteristics, and group characteristics into services. To meet the requirements of the intended model, a fourth step includes mapping services and CTI formats.

Those four steps are explained in more detail in the following: (a) *Identify subset of SOC services*, was conducted with the help of literature analysis. Thereby, all publications dealing with SOC capabilities can be considered. In the methodology of [Nickerson et al. \(2013\)](#), capabilities are referred to as objects (as they are not SOC specific). Here, no classification is performed, so the capabilities are available in different degrees of abstraction and can overlap in some cases. (b) *Identify common characteristics*, aims to find properties of the identified SOC capabilities. Thereby, the characteristics that differ across the capabilities and thus enable classification are essential. This step is, to some extent, carried out intuitively, as an objective identification of the properties is not always possible. Based on this, (c) *Group characteristics into services*, is the next operation. The groups form the SOC services, whereby an umbrella term must be identified for each group. Finally, (d) *Map to CTI formats*, connects services by comparing their respective goals and areas of application. For CTI formats, multiple assignments are possible since individual CTI formats can affect several SOC services.

Phase 5, conception of transfer and evaluation, is conducted by adapting standard procedures to our specific requirements. For this, some possibilities are already given by [Becker et al. \(2009\)](#), which can be used as established means. To enable a targeted approach, we first define the requirements for transfer and evaluation and, building on this, carry out phases 6 and 7.

Phase 6, implementation of transfer media, describes the development of a prototypical tool that an employee of an organization can use to perform a maturity assessment of their

SOC. Thereby, particular attention is paid to intuitive applicability and an appealing visual presentation of capability and maturity levels.

Phase 7, evaluation, investigates the relevance and applicability of the proposed maturity model. Evaluation is achieved in two ways: 1) a user study is conducted to illustrate the relevance of the defined problem, and 2) in-depth expert interviews demonstrate the model's practical applicability and relevance.

Phase 8, rejection of maturity model, is about constantly assessing if the maturity model still fits. It re-confirms the results of the evaluation. If the results do not meet the requirements, a new design and development iteration is carried out. However, eventually, a model must be rejected as ever-changing requirements make it impossible to adapt. We list this phase for completeness. Please note that neither reassessment nor rejection due to missing adaptation applies to the iteratively developed model within our work.

5. Integrated SOC capability maturity model

Within an integrated SOC capability maturity model, the SOC concept is accompanied by inter-connected domains. We opt for this approach with an additional focus on CTI due to previously described overlaps between the organizational concepts SOC, CSIRT, and the use of CTI for security operations. In the following, we compare related maturity models specifically targeting SOC, CSIRT, Incident Response (IR), and CTI. We then derive a three-tier capability maturity model architecture. From highest tier to lowest tier, maturity levels, SOC services, and CTI formats specify the integrated model.

5.1. Related maturity models

Organizations seek guidance on how to build, assess and improve capabilities bundled within a SOC. Guidelines and recommendations published in recent years aim to provide details on the many aspects of consideration (Taurins, 2020; Zimmerman, 2014). Also, SOC capability and maturity models have been developed. Towards an integrated and CTI-focused SOC capability maturity model, we address the first research question about the absence or existence of specific maturity models by searching and examining related maturity models. Our initial search was conducted in early 2021 and includes peer-reviewed academic literature and gray literature on maturity models. Existing maturity models can broadly be classified into two groups. The first group includes models proposed by academia, both peer-reviewed and non-peer-reviewed. The second group includes maturity models proposed by organizations and special interest groups in information security. We indicate the origin of a given maturity model according to the two groups. As SOC and incident response typically comprise aspects of CTI, we cover these models before exclusive CTI-related models.

Initiated in 2016, SOC-CMM by Van Os (2016) is a SOC-centered capability maturity model based on a scholarly study. Besides the current version SOC-CMM 2.1 and its MS-Excel self-assessment tool, there exists a separate model adapted to the needs of incident response teams. SOC-CMM covers

the domains of business, people, process, technology, and services in detail. In contrast to SOC-CMM, we argue that CTI is not only a separate service but builds the basis for a variety of SOC services. Therefore, we emphasize SOC services, CTI dependencies, and the intelligence-driven underlying of security operations and incident response. Another extension to SOC-CMM is introduced within the academic work of Acartürk et al. (2020). The authors conclude that generic continuous improvement methods from the field of quality management can be supportive elements.

Based on the results of a comparative study on generic cybersecurity capability maturity models (Rea-Guaman et al., 2017), we take the *Cybersecurity Capability Maturity Model (C2M2)* into account. As C2M2 is cybersecurity-oriented and targeted at an organizational environment, it relates to core SOC characteristics. C2M2 is a collaborative product of the US Department of Energy and Carnegie Mellon University and includes ten domains (e.g., threat and vulnerability management) for which objectives are defined (Christopher et al., 2014). When we define SOC services, these domains are assessed and either aligned or excluded based on their granularity and fit.

A prominent maturity model for CSIRT is the *Security Incident Management Maturity Model (SIM3)* introduced by Stikvoort (2015). SIM3 is a maturity model and online self-assessment tool currently provided by the non-profit Open CSIRT Foundation (OCF). Organizations within the incident response community use SIM3 to certify organizational incident management. While covering essential capabilities in the domains of tools and processes, we identify a gap within SIM3 regarding the systematic use of CTI for organizational processes.

Maturity assessment of incident response with a lesser focus on organizational integration is addressed by the non-profit CREST (The Council for Registered Ethical Security Testers). The *Cyber Security Incident Response Assessment Tool (CSIR-MAT)* covers basic assessment for three phases: prepare, respond, and follow-up. However, CREST also provides the *Cyber Threat Intelligence Maturity Assessment Tool (CTI-MAT)*. Here, threat intelligence is structured according to a life cycle from direction to review. We find that the processing and dissemination operations in CTI-MAT include CTI formats.

Other maturity models – CTI-CMM (Lourenco, 2018) and CTIM (Luchs and Doerr, 2020) – are outlined by ENISA, and by researchers at Hasso Plattner Institut and TU Delft respectively. Both introduce a CTI life cycle focusing on data and its categorization. We identify maturity levels and indicators valuable for maturity assessment.

Finally, an academic proposal towards a CTI maturity model is introduced by Sillaber et al. (2018). The high-level maturity model is derived from previously conducted expert interviews and focused on inter-organizational CTI sharing. Application scenarios for CTI play a role for more mature organizations. Thus, we consider the use of CTI formats a reasonable condition for maturity improvement.

Foundations. All related maturity models adhere to the standard multi-tier structure with different maturity levels, different capability levels, or both. Besides, maturity models for SOC, CSIRT, or incident response typically cover people, processes, and technology. We conclude that these foundations provide both guidance and flexibility for develop-

ing an integrated and CTI-focused SOC capability maturity model.

5.2. Design decisions

Capability maturity models consist of standard components and have specific characteristics. Therefore, the development of a CMM is accompanied by multiple design decisions. In the following, we outline the design decisions of our model, which will be detailed in the subsequent sections of this paper. Please note that the CMMI, the methodology by Becker et al. (2009), and the arguments below guide our design decisions.

Objective. CMMs fulfill one or more specified objectives. The objective of our model is the development of a mature, intelligence-driven SOC. We reason that as data and its analysis are key to successful business operations (business intelligence), the same holds for security operations (threat intelligence).

Scope. CMMs are defined by their scope. The scope of our model is the operationalization of CTI in SOC. We focus on organizational CTI capabilities required for SOC services. Consequently, we build on widely used CTI formats as these are crucial for understanding and using CTI.

Users. CMMs have a specified target audience. Our model is aimed at SOC and information security personnel. Most CMMs address managerial positions with executive powers. We envision SOC managers, SOC consultants, and CISOs to apply our model within organizations.

Tiers. CMMs consist of hierarchically structured elements for which we use the term *tiers*. From top to bottom, maturity levels, capabilities and capability levels, and indicators represent the three standard tiers in CMMI. We define four maturity levels to capture SOC maturity concerning CTI. Therefore, we adapt the CMMI naming convention. Capabilities in our model are represented by six SOC services identified in the literature. The decision for SOC services as capabilities is influenced by the model's scope. Our model further includes six capability levels to show the implementation of a given SOC service. Capability levels are closely linked to indicators. CTI formats serve as indicators and, mapped to the SOC services, determine capability levels. The design decision for CTI formats is based on the assumption that organizational use of CTI involves formats. Thus, indicative questions based on CTI focal points address the degree of CTI format coverage and represent the capability levels.

Mapping. CMMs must deal with mapping the individual tiers. At the center of our model, we tie CTI formats and SOC services together and thus realize mapping indicators to capabilities. We leverage the well-known NIST incident response life cycle to map capabilities and their levels to maturity levels. We argue that SOCs aim for complete life cycle coverage. However, as resources are scarce, SOCs will improve CTI maturity step-by-step and start with preventive measures. Also, for the degree of CTI format consideration, we assume a successive approach with source, quality, and integration of CTI formats elementary for organizational use and backed by CTI literature.

Approach in a nutshell. In short, our model is based on how well organizations handle CTI formats. Mapping CTI for-

formats to SOC services allows determining a capability level for these services. Then, all SOC services combined determine the intelligence-driven maturity of the SOC (see Fig. 3).

5.3. Architecture

Having discussed related maturity models, we determine a maturity model development strategy according to the procedure put forward by Becker et al. (2009). As existing maturity models already provide an initial setup, we dismiss the option to develop an entirely new capability maturity model. Instead, we integrate and refine existing elements and direct focus on the area of CTI and CTI formats. The decision for CTI formats is influenced by the significance of the CTI concept for operational cybersecurity (Brown and Lee, 2021; Shin and Lowry, 2020) and formats required as its core. We further reject the option to use the industrial Software Process Improvement and Capability Determination (SPICE) methodology (Dorling, 1993). This decision in favor of the CMMI approach is influenced by the scientific prevalence of CMMI and its more general direction than SPICE's engineering aspects.

CTI-SOC2M2. The architecture of our CTI-focused model, referred to as **CTI-SOC2M2**, is visualized in Fig. 3. From left to right, the architecture consists of three tiers – CTI formats, SOC services and the associated capability levels, and maturity levels for the intelligence-driven SOC.

On the lower tier of CTI-SOC2M2, threat intelligence focus is realized via *CTI formats*. The CTI formats in this tier function as an indicator and directive for capability fulfillment, eventually leading to maturity assessment. CTI formats being part of the CTI concept represent organizational CTI capabilities as they link CTI artifacts and application domains. In the context of the CTI-SOC2M2, the CTI formats are assigned to SOC services.

On the central tier, we organize *SOC services* representing procedural elements. CTI-SOC2M2 is a capability maturity model centered exclusively on services. We opt for this approach because SOC services are integral to the operationalization of CTI. Nevertheless, it is worth mentioning that complementary maturity models and frameworks (e.g., ISO 27001) covering governance and people exist. As SOC services use technologies, where applicable, we point to relevant dependencies.

Both the central tier and the lower tier document the decision for an integrated maturity model. As it can be observed that maturity models in highly-specific domains exist (e.g., Digital Forensic Readiness (Engbrecht et al., 2020)), we aim to combine detailed elements of CTI and SOC. Therefore, we map CTI formats to SOC services allowing capability assessment for SOC services via the CTI formats. CTI, in general, has the benefit of introducing external insights into threats not (directly) visible within an organization. Besides, we also aim to provide enough differentiation from generic information security maturity models by integrating an extensive SOC study's research results.

On the upper tier, *maturity levels* indicate the current state of SOC maturity with regard to CTI. These maturity levels are dependent on the capability levels reached by individual SOC services and guide step-wise improvement.

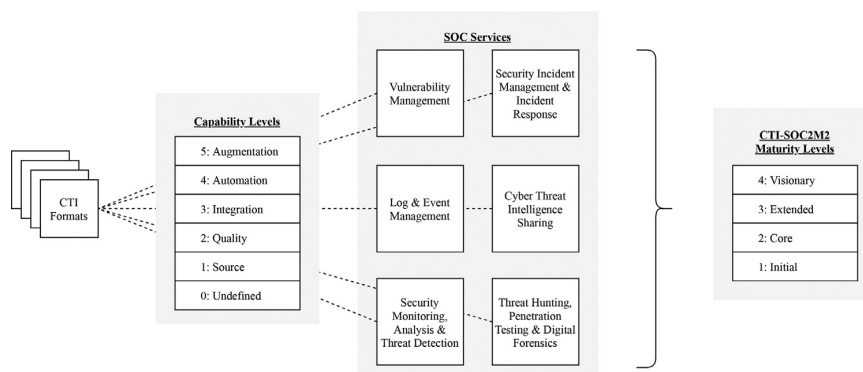


Fig. 3 – CTI-SOC2M2 Architecture.

Table 2 – Related maturity models.

Model	Documentation	Category
SOC-CMM	Van Os (2016)	SOC
SOC-CMM*	Acartürk et al. (2020)	SOC
C2M2	Christopher et al. (2014)	SOC
SIM3	Stikvoort (2015)	CSIRT
CSIR-MAT	CREST (2014)	IR
CTI-MAT	CREST (2016)	CTI
CTI-CMM	Lourenco (2018)	CTI
CTIM	Luchs and Doerr (2020)	CTI
CTI-MM	Sillaber et al. (2018)	CTI

Table 3 – CTI formats and frameworks.

CTI Category	CTI Format
Threat Report	IODEF
	MISP
	STIX
	TAXII
	VERIS
	CACAO
Course of Action (CoA)	OpenC2
	RE&CT
	CAPEC
Tactic, Technique, Procedure (TTP)	Cyber Kill Chain
	Diamond model
	MITRE ATT&CK
	CVSS
Scoring System	CPE
	CWE
	CVE
Indicator of Compromise (IoC)	GENE (logs)
	OpenIOC
	Sigma (logs)
	Snort (traffic)
	YARA (files)
	Zeek (traffic)

From a functional perspective, CTI-SOC2M2 is based on qualitative and quantitative measures. Qualitative assessment of CTI formats leads to a capability level for the mapped SOC service. It is worth mentioning that organizations can deem CTI formats not applicable for their context and omit these from qualitative capability assessment. Thus, high capability levels can be achieved in our model by considering only a few selected CTI formats. In contrast, the overall intelligence-driven SOC maturity is based on a quantitative approach. Therefore, capability levels of different SOC services are considered and assessed (later discussed in Section 6.2).

5.4. CTI formats

Before mapping CTI formats to SOC services, an over-view of relevant formats and frameworks is displayed in Table 3. We base our selection of these formats and frameworks on the influential work by Dandurand et al. (2014), consider promising new format developments, and ignore deprecated formats. Format categorization is based on generic CTI elements (Mavroidis and Bromander, 2017).

IoC formats are often closely related to defensive security systems such as SIEM tools, IDS, or IPS. Their primary purpose is the identification of rules or patterns within logs, network traffic, or files. Security enumerations are used to identify relevant artifacts (e.g., IT assets or vulnerabilities). A systematic approach to assist decision-making is described by scoring systems. While TTP formats cover mainly attacker behavior

and the offensive side, CoA formats describe procedural countermeasures conducted as a method of defense. Finally, encompassing threat report formats handle aggregated CTI and include aspects of other CTI formats.

5.5. SOC services

SOC services are the foundation for SOC operation. We use primary data from a previous SOC study (Vielberth et al., 2020) for the definition and iterative development of SOC services. The data used is the result of a structured literature review. The literature corpus covers 158 publications on SOC in general⁶. We only consider parts of the literature helpful and thus explicitly cite only highly relevant works. However, we want

⁶ <https://ieeexplore.ieee.org/ielx7/6287639/8948470/9296846/supp1-3045514.xlsx?arnumber=9296846>.

Table 4 – SOC field studies and identified SOC services.

Reference	SOC Services
Kowtha et al. (2012)	Log & Event Management Analysis Security Incident Management Incident Response CTI Sharing Threat Hunting
Jacobs et al. (2013)	Vulnerability Management Log & Event Management Security Monitoring Analysis Incident Response Threat Hunting Penetration Testing
Onwubiko (2015)	Vulnerability Management Log & Event Management Analysis Threat Detection Incident Response CTI Sharing Digital Forensics
Settanni et al. (2017)	CTI Sharing CTI Usage

point to details and supplementary material of the literature review, which we use as starting point.

SOC elements of varying granularity can be categorized into topic areas. This categorization is of particular interest for the clustering of SOC services and CTI-SOC2M2 development. We perform an in-depth examination of publications that cover the two topic areas capability or maturity. Outlined SOC processes (e.g., Kowtha et al. (2012); Schinagl et al. (2015)) provide further guidance for the development of SOC services.

This knowledge is then combined with the methodology by Nickerson et al. (2013) to form clusters which we name SOC services. The approach by Nickerson et al. (2013) is particularly suitable as our classification is essentially a single-layer taxonomy. Thereby, it pays special attention to mutual exclusiveness, which, according to de Bruin et al. (2005), is a major requirement for the defined services. Besides, we use existing and sufficiently documented SOC maturity models (e.g., SOC-CMM (Van Os, 2016)) to iteratively validate our SOC services. Mapping of SOC services and CTI formats is then assisted by core aspects defining each SOC service and work on CTI format categorizations (Hernandez-Ardieta et al., 2013).

Even though the majority of the papers in the literature corpus deal with SOC services, SOC is usually not considered holistically. Thus, often only selected sub-areas or sub-services are analyzed. In particular, there is a strong focus on log & event management and analysis. Table 4 lists the most important field studies that take a holistic view of a SOC, as these are particularly important for identifying SOC services. Within the table the identified SOC services within these studies are listed.

5.5.1. Vulnerability management

Existing vulnerabilities define threats to information systems and IT infrastructure. Vulnerability management is part of

different maturity models, industry standards, and SOC research (Farris et al., 2018). However, as vulnerability management deals with an adequate handling of known vulnerabilities, it is a SOC service influenced by CTI (Chismon and Ruks, 2015). Therefore, maturity assessment must consider relevant CTI formats and sources such as exploit and vulnerability databases. Applicable CTI formats to vulnerability management cover both security enumerations and scoring systems. They provide a common understanding, reference, and assessment of vulnerability severity to guide decision-making. Additionally, vulnerabilities relate to IT assets.

CTI formats:

- CPE – Common Platform Enumeration allows reference and identification of classes of IT assets. In its current version 2.3., CPE is maintained by the National Institute of Standards and Technology (NIST). Each IT asset's characteristics are described via string-based format (Cheikes et al., 2011).
- CVE – Common Vulnerabilities and Exposures is a security enumeration to refer to vulnerabilities in IT assets uniquely. It is maintained by MITRE and a community. Its data constitutes the basis for the US National Vulnerability Database (NVD) (Baker et al., 1999).
- CVSS – Common Vulnerability Scoring System version 3 provides a formal procedure to specify the severity of a vulnerability ranging from 1 to 10. It is maintained by FIRST and can be applied in different modes, including a primary assessment and consideration of organizational and environmental factors (Forum of Incident Response and Security Teams (FIRST), 2019).

Referring to the motivational example, the latest version of the Microsoft Exchange Server affected is specified as `cpe:2.3:a:microsoft:exchange_server:2019:cumul._update_8` and CVE-2021-26855 is CVSS-rated 9.8.

5.5.2. Log and event management

Logs and events capture information about system processes and system states. As a consequence, log and event management is concerned with internal data required for security analysis. Being part of various industry standards, IT operations, and SIEM tools, this service is essential for SOC (Madani et al., 2011). But, to conduct effective log and event management, external CTI can foster security assessment and alignment to security goals. It is thus necessary to consider data formats describing attacker behavior documented by logs and system events. These data formats go beyond the essential log formats such as Syslog (Gerhards et al., 2009), NCSA (Apache HTTP Server Project, 1995), EVTX (Microsoft, 2018), or Common Event Format (CEF) (ArcSight, 2010) and describe threat detection patterns.

CTI formats:

- GENE - Go Evtx sigNature Engine and rule format version 1.6 aims to provide signatures for Windows event logs. The open-source format proposed in 2018 by RawSec company centers on JSON-described rules (RawSec - Quentin Jerome, 2018).

- Sigma – Generic Signature Format for SIEM Systems is an open-source format aimed at log files and log events. The project driven by [Roth and Patzke \(2017\)](#) includes a YAML-based format specification to describe threat identifiers and allow detection.

Referring to the motivational example, a Microsoft Exchange Server captures its logs and events in .evtx-files. Using GENE might be a feasible approach to determine anomalies.

5.5.3. Security monitoring, analysis & threat detection

Security monitoring is a continuous approach to ensure an organization's security goals. At the center of a SOC, security monitoring copes with an aggregate view of IT assets and their security ([Onwubiko, 2015](#)). In conjunction with security monitoring, security analysis and threat detection can yield additional insights into specific security aspects and identify threats. While it is possible to conduct security monitoring, analysis, and threat detection without threat intelligence, the general threat landscape can provide essential clues. Contrasted with CTI on current malware, command and control servers, and ongoing cyber attacks, variations witnessed in network traffic and system behavior allow organizations to initiate appropriate follow-up steps. CTI formats applicable for this SOC service mainly include information on IoC. As an example for additional CTI, we also list the MITRE ATT&CK framework and the comprehensive STIX format.

CTI formats:

- OpenIOC – The OpenIOC format allows description of different types of IoCs. Developed by Fireeye (formerly Mandiant) in 2013 the current schema version 1.1 is XML-based, open-source, and has a criteria section to match specified values against, for example, files or processes ([Ross et al., 2013](#)).
- Snort – The Snort format provides rules for detecting network traffic threats in combination with the open-source IDS/IPS tool. Snort is maintained by Cisco Talos and a community. Its rules are based on a custom schema and describe actions and detection parameters ([Snort Team, 2021](#)).
- Zeek – The Zeek signature format for network traffic supports matching threat patterns. The format is part of the open-source Zeek (formerly Bro) network security monitoring tool, which contains additional components. Maintenance is realized by a community ([The Zeek Project, 2021](#)).
- MITRE ATT&CK – The ATT&CK framework categorizes and details adversary behavior with tactics, techniques, and mitigation procedures. Maintained by MITRE, the framework evolved and comprises both information for IT and OT environments ([Strom et al., 2018](#)).
- STIX – In version 2.1, Structured Threat Information Expression (STIX) is a comprehensive CTI format capturing low-level cyber-observables and information on TTPs, CoAs and their dependencies. Initiated in 2012, STIX is maintained by OASIS and centers on JSON-based threat reports ([OASIS Cyber Threat Intelligence \(CTI\) Technical Committee, 2020a](#)).
- see also CPE, CVE, CVSS, and Sigma.

Referring to the motivational example, available OpenIOC-based indicators for malware assist detection of a Microsoft Exchange Server compromise. Other malicious IP addresses enable alerts in IDS and are grouped in STIX2.1 threat reports.

5.5.4. Threat hunting, penetration testing & digital forensics

Threat hunting, penetration testing, and digital forensics are all concerned with detailed investigations. In-depth analyses aggregated in this SOC service go one step further than security monitoring and aim to find evidence of ongoing attacks, malware, existing vulnerabilities, and procedural deficiencies. As it is common practice to conduct threat hunting, penetration testing, and digital forensics to test actively and identify incidents ([Hámornik and Krasznay, 2018](#)), these activities rely on information. While it is necessary to resort to internal information, this is often not sufficient. However, the use of external CTI can integrate typical attack patterns and other identifying elements. Therefore, CTI formats describing TTPs, software weaknesses, and IoCs are of relevance. Together with appropriate technology, comprehensive CTI formats such as MISP can further assist specific actions.

CTI formats:

- YARA – The YARA rule format allows to describe patterns to match against files. Developed at VirusTotal, the open-source YARA tool and format support detection of malicious files. Community driven open-source rule repositories exist ([VirusTotal - Victor Alvarez, 2014](#)).
- CWE – Common Weakness Enumeration is focused on software flaws. It is maintained by MIRE and lists software weaknesses by three categories (i.e., software development, hardware design and research concepts) ([MITRE, 2020](#)).
- CAPEC – Common Attack Pattern Enumeration and Classification is used to refer to attack patterns. Maintained by MITRE, CAPEC is focused on common application weaknesses and categorizes patterns by mechanisms and domains of attack ([CAPEC Team, 2020](#)).
- Cyber Kill Chain – Various cyber kill chains exist. For example, the Lockheed Martin cyber kill chain describes various stages of an attack and thereby assists detection and defense ([Hutchins et al., 2011](#)).
- Diamond model – The diamond model supports intrusion analysis with four adjacent categories: adversary, capability, victim, and infrastructure. These define core characteristics of an adversary and its campaign ([Caltagirone et al., 2013](#)).
- MISP – Open Source Threat Intelligence Platform and format centers on events, attributes and tags to comprehensively describe threat intelligence. The open-source project supported by Computer Incident Response Center Luxembourg (CIRCL) and the European Union allows CTI collection and sharing ([Wagner et al., 2016](#)).
- see also CVE, OpenIOC, Snort, and MITRE ATT&CK.

Referring to the motivational example, analysis of other attack campaigns by threat actors exploiting the Microsoft Exchange Server vulnerability is relevant. Analysis and threat hunting can start with threat actor Hafnium and its use of web shells.

5.5.5. Security incident management & incident response

A security incident may have various causes potentially leading to harm for an organization. As a security incident is a type of event that violates security policies, it is essential to manage security incidents and respond with appropriate measures. One aspect of managing is incident triage leading to a prioritization of actions based, for example, on impact or available resources (Shah et al., 2019). The greater concept of security incident management and incident response is currently gaining momentum. A variety of dedicated Security Orchestration, Automation and Response (SOAR) systems (Neiva et al., 2020) and the underlying incident response standardization aim to establish a more efficient approach. Besides, for incident response training, cyber ranges are discussed. Thus, combined with existing ticketing systems, it becomes necessary to consider CTI formats centering on CoAs to support these use cases.

CTI formats:

- CACAO – Collaborative Automated Course of Action Operations for Cyber Security format version 1.0 is centered on incident response workflows. CACAO is maintained by OASIS. It is capturing information about procedural logic and actions in JSON-based playbooks and supports sharing (OASIS, 2021).
- OpenC2 – Open Command and Control format version 1.0 represents commands for incident response machine-to-machine communication. Maintained by OASIS, granular OpenC2 actions are JSON-based and transferred to defensive systems for execution (OASIS, 2020).
- RE&CT – The RE&CT framework includes a matrix representation of incident response stages and actions. Besides, the ATC project behind RE&CT introduced YAML-based playbooks (ATC Project, 2020).
- see also CVE, OpenIOC, STIX, and MISP.

Referring to the motivational example, integrating the PowerShell script into a CACAO playbook is beneficial for the incident response workflow. IP addresses belonging to Command-and-Control infrastructure can be blocked using a firewall and initiating an outbound traffic re-direct with OpenC2 message.

5.5.6. Cyber threat intelligence sharing

Cyber Threat Intelligence is not only part of other SOC services but also a SOC service of itself. Based on incident reporting, the Cyber Threat Intelligence sharing service copes with comprehensive threat reports. Ensuring adequate gathering of external information and internal dissemination is at the center of this SOC service. Nevertheless, using CTI demands comprehensive and structured CTI formats which encapsulate relevant threat information. Therefore, threat report formats are included to build a knowledge base about cyber attacks, threats, and security incidents. The sharing aspect of CTI is also incorporated in threat report formats focusing on data transfer.

CTI formats:

- IODEF – Incident Object Description Exchange Format version 2 supports the representation and the exchange of security incident reports and indicators. The IETF standard

is based on XML and includes objects for different types of CTI (Danyliw, 2016).

- TAXII – Trusted Automated eXchange of Indicator Information format version 2.1 is used for transferring and collecting STIX-based threat reports. Associated with STIX and maintained by OASIS, the framework includes services, HTTPS transport, and client-server architecture to facilitate the sharing of CTI (OASIS Cyber Threat Intelligence (CTI) Technical Committee, 2020b).
- VERIS – Vocabulary for Event Recording and Incident Sharing version 1.3.2 enables incident description based on the categories actors, actions, assets and attributes. Developed by Verizon, it is open-source and uses JSON to represent CTI. A VERIS Community Database (VCDB) includes public security incidents (VERIS Community, 2021).
- see also Sigma, YARA, STIX, and MISP.

Referring to the motivational example, using a TAXII server to query CTI provides fast access and supports following SOC services. Also, IODEF and VERIS can document security incidents and build a historical database with both high- and low-level incident descriptions.

6. Maturity assessment with CTI-SOC2M2

The use of CTI-SOC2M2 for maturity assessment is based on CTI formats that serve as indicators. These indicators and indicative questions allow organizations to self-assess their current capability and maturity level and show steps towards improvement.

6.1. CTI formats and capability levels

Typically, capability maturity models use indicative questions to determine capability fulfillment and the associated capability level. We follow a generic and qualitative approach applicable to all individual CTI formats and SOC services. As the indicative questions for capability levels pertain directly to CTI and CTI formats, similarities with maturity models designed solely for CTI exist. It is also worth mentioning that capability levels are built upon each other. Lower levels, e.g., *Source* and *Quality*, are necessary requirements to reach the next capability level (e.g., *Integration*). After selecting applicable CTI formats according to the specific organizational setting, the following categories and indicative questions must be answered to assess any given CTI-based SOC service.

Capability Levels:

- 0: Undefined** – CTI and CTI formats have not yet been considered.
- 1: Source** – Have you determined and assessed the source of CTI with the mentioned CTI format(s)?
- 2: Quality** – Have you applied appropriate measures to assess the quality of the CTI structured with the mentioned CTI format(s)?
- 3: Integration** – Have you integrated CTI and the mentioned CTI format(s) into your organizational processes and technology architecture?

- 4: **Automation** – Have you automated retrieval, use and dissemination of CTI based on the mentioned CTI format(s)?
- 5: **Augmentation** – Have you set-up a monitoring mechanism to cope with new developments within CTI and new CTI format(s)?

We determined the capability levels and indicative questions by considering focal points of the CTI concept. Previous studies emphasized the importance of CTI sharing platforms and the source of CTI (Bauer et al., 2020; Bouwman et al., 2020). Besides, the quality of CTI and the expressiveness of CTI formats are highly relevant (Li et al., 2019; Schaberreiter et al., 2019; Schlette et al., 2021). SOC services are specified by more granular processes, which must handle the integration of CTI, its formats, and technology. Integration builds a prerequisite to fully achieve effectiveness via automation. Towards the ultimate goal of security orchestration, automation, and incident response, CTI is one essential element (Islam et al., 2019). However, the current developments show that the state-of-the-art of CTI is constantly shifting (Brown and Lee, 2019). Therefore, it is necessary to monitor and continuously extend the organizational understanding of CTI formats and associated concepts.

6.2. SOC services and maturity levels

SOC services are assessed based on the CTI formats and indicative questions mentioned in Section 6.1. Transitioning from SOC services and capability levels to overall CTI-SOC maturity levels demands a methodology outlined below.

Maturity Levels:

- 1: **Initial** – Capability level 2 is reached for Log & Event Management, Security Monitoring, Analysis & Threat Detection and Vulnerability Management.
- 2: **Core** – Capability level 2 is reached for Security Incident Management & Incident Response and Cyber Threat Intelligence Sharing. All previous services reached capability level 3.
- 3: **Extended** – Capability level 2 is reached for Threat Hunting, Penetration Testing & Digital Forensics. All previous services reached capability level 3.
- 4: **Visionary** – Capability level 4 is reached by all SOC services.

We first define four maturity levels: *Initial*, *Core*, *Extended*, and *Visionary*. The naming of these maturity levels indicates SOC functionalities addressed by CTI. Improvement of CTI-SOC maturity within an organization depends on the individual fulfillment levels for the SOC services. As CTI-SOC2M2 adheres to a step-wise approach, maturity levels are downgraded if underlying SOC service capabilities cease to exist.

Our methodology for CTI-SOC2M2 is inspired by the NIST Incident Response Life Cycle (Cichonski et al., 2012). Reaching higher maturity levels is equivalent to addressing more aspects of the Incident Response Life Cycle more thoroughly (see Fig. 4). Whereas organizations aim to implement all the individual aspects of the incident response life cycle, we envision organizations with limited resources and new to the concept of CTI and CTI formats to approach CTI-driven SOC maturity

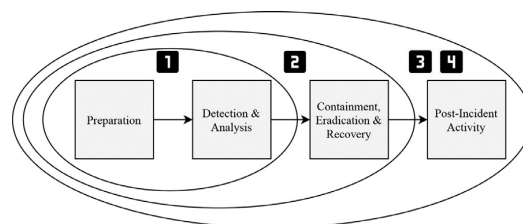


Fig. 4 – Maturity level scope compared to NIST Incident Response Life Cycle (Cichonski et al., 2012).

step by step. Consequently, an organization might not cover all its aspects concerning CTI formats but can still have some generic measures in place. Also, organizations might decide to remain on a specific maturity level due to the delegation of certain SOC services.

Therefore, with *Log and Event Management*, *Security Monitoring*, *Analysis & Threat Detection* and *Vulnerability Management* on capability level 2: *Quality* an initial maturity level is reached and likewise aspects of preparation, detection and analysis of the incident response life cycle covered by CTI formats. Progress towards core maturity is possible when the aforementioned SOC services reach level 3: *Integration* and additionally *Security Incident Management & Incident Response* and *Cyber Threat Intelligence Sharing* reach level 2: *Quality*. This is accompanied by covering containment, eradication & recovery of the incident response life cycle with CTI formats. Including *Threat Hunting*, *Penetration Testing & Digital Forensics* on level 2: *Quality* as well as progressing the other SOC services towards 3: *Integration* leads to an extended CTI-SOC. This further implies covering post-incident activity of the incident response life cycle in detail. Finally, starting with at least capability level 4: *Automation* for all SOC services the visionary maturity level is reached. Aspects of the incident response life cycle are advanced and additional progress with 5: *Augmentation* for SOC services is still possible.

Motivational Example. We want to document the CTI-focus of an illustrative SOC which reached the *extended* maturity level and emphasize aspects of the motivational example introduced in Section 2. While an acceptable maturity is already reached with the *core* maturity level, this illustrative SOC covers all SOC services with at least integrated CTI formats. Concerning the Microsoft Exchange Server breach, the organization witnessed a compromise but has a sufficiently integrated log and event management where the public Sigma rule⁷ has been analyzed and is part of the organizational CTI process for SIEM systems. Security monitoring, analysis, and threat detection integrate SIEM systems, IDS, and firewalls with CTI formats leading to the detection of network traffic to IP 218.103.234.[.]104 also listed in a queried STIX2.1 threat report. Vulnerability management covers processes considering newly published CVE-IDs. This is aimed to avoid missing other related vulnerabilities (e.g., CVE-2021-27065). NVD is actively and regularly searched, compared to CVEs in threat re-

⁷ https://github.com/SigmaHQ/sigma/blob/master/rules/web/web_exchange_exploitation_hafnium.yml.

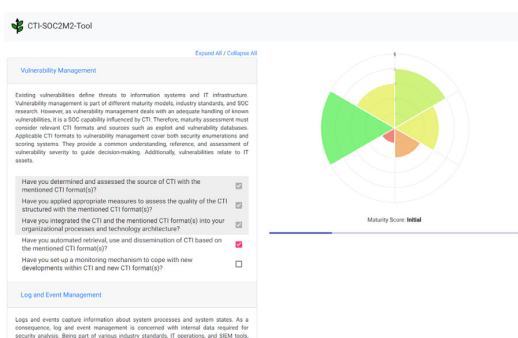


Fig. 5 – CTI-SOC2M2 self-assessment tool with radar chart for SOC services.

ports, and complemented with other vulnerability databases (e.g., OSV - Open Source Vulnerabilities⁸). For CTI sharing, the organization hosts its own MISP instance to share security information between the headquarter and branches. The security incident management and incident response SOC service is triggered by the high severity indicated by the CVSS score and focused on web shell removal. A CACAO playbook is used to define and structure the granular actions. A suspicious file `shell.aspx` was submitted to OTX, deleted, and the Microsoft Exchange Server patched. Threat hunting, penetration testing, and digital forensics were then conducted for in-depth analysis. Based on the different types of web shells the SHA-1 hash `eb8d39ce08b32a07b7d847f6c29f4471cd8264f2` was found and its actions analyzed. As a result, a YARA rule was adapted.

With SOC services this mature, the illustrative organization avoided the exfiltration of large amounts of data and a widespread lateral movement by attackers on the organizational networks. In this particular case, patches became available only after the first successful exploits by threat actors had been conducted. As a result, even a mature CTI-driven SOC could not prevent the initial attack. However, as response measures were applied promptly, further negative consequences could be limited. In addition, risk management was always aware of the ongoing operations and could advise decisions.

6.3. Prototypical implementation

The presented maturity model is based on the assumption that employees within an organization record the capability level for each SOC service. In this scenario, self-assessment is best supported with a suitable tool. Essentially, the tool provides two functions. On the one hand, it can be used to record capability levels. On the other hand, it will calculate and display the overall SOC maturity based on the methodology outline in Section 6.2.

Fig. 5 depicts the structure of the prototypical CTI-SOC2M2 self-assessment tool. A demo version of the implementation can be accessed online (<https://antumin.github.io/>

⁸ <https://osv.dev/>.

CTI-SOC2M2/). The tool layout is divided into two parts. On the left-hand side, the CTI-based capability levels for each SOC service can be recorded. For this purpose, a description is intended to help the user understand the SOC service characteristics. A drop-down menu then allows the user to select a capability level referring to CTI and CTI formats. On the right-hand side, the maturity level is displayed. The maturity level of the assigned to the overall SOC is stated, and a radar chart visualizes the capability breakdown for the SOC services. This visualization enables users to immediately identify deficient SOC services and improve CTI efforts to progress towards a more mature SOC.

From a technical perspective, the tool is implemented as a web app. This decision allows for platform-independent use independent from other commercial software such as Microsoft Excel, typically used for maturity models. The web app was developed using the Angular⁹ framework, based on HTML, Javascript, and CSS. The source code is open-source and published on GitHub¹⁰ enabling further development and future research.

7. Evaluation

This section concludes the methodology of capability maturity model development outlined earlier. We use a mixed-method approach, combining a quantitative user study with a qualitative evaluation based on expert interviews. With the two components, we aim to document relevance and applicability.

User study

To show the relevance of SOC maturity and its threat intelligence focus, we conducted an international user study.

Design & Procedure: The impact of a security analyst's CTI-based skills on the ability to detect attacks is explored with three phases:

1. *Assessment of pre-knowledge and attack detection skills:* During the first phase, participants are asked questions using a questionnaire that measures their pre-knowledge of CTI formats. In addition, the participants are asked how accurately they can detect attacks and how extensive their attack detection knowledge is.
2. *CTI-based SOC training:* The second phase forms a training session. Participants learn to understand a CTI format for describing detection rules and indicators of compromise using dSIEM¹¹, an open source SIEM system based on Elasticsearch¹². For this purpose, introductory videos and texts are provided. The SIEM system is made available and data from real attacks is inserted.
3. *Assessment of post-knowledge and attack detection skills:* In order to verify the effect of the training on the participants

⁹ <https://angular.io/>.

¹⁰ <https://github.com/antumin/CTI-SOC2M2>.

¹¹ <https://www.dsiem.org/>.

¹² <https://www.elastic.co/>.

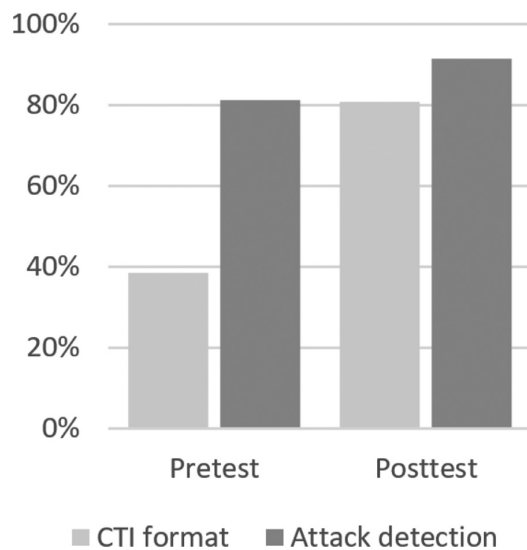


Fig. 6 – Participants' knowledge on CTI formats and attack detection before and after the user study.

skills to detect attacks, the assessment with the questionnaire from phase 1 is conducted a second time.

The user study was conducted at a German and a Greek university with $n = 44$ participants. All of them were students with an IT security background. 22 participants were Greek and 22 German, whereby 12 female and 32 male students participated. 20 students were undergraduate, and 24 postgraduate. The complete data set of the user study can be found as public data on GitHub (<https://github.com/DigitalTwinSocCyberrange/userStudy>).

Results: During phase 1 and phase 3, the participants were asked eight questions to assess their knowledge about CTI formats and their skills in detecting attacks. In Fig. 6 the percentage of correct answers during phase 1 (pretest) and phase 3 (posttest) are shown. In order to statistically show the increase in knowledge, a t-test was conducted. The user study showed that the mean of correctly answered questions about CTI formats significantly increased by 42.05% ($t = -3.448$, $SD = 0.331368$, $p < .001$). Additionally, a significant mean increase of 10.23% ($t = -3.448$, $SD = 0.196763$, $p = .0013$) considering attack detection can be observed. The t-test shows that the training had a significant positive effect on both variables. We conclude that training on CTI formats can have a positive effect on the overall attack detection knowledge. This fact is particularly interesting as we base our approach on a qualitative assessment of CTI formats, leading to SOC service capability levels.

Expert interviews

To validate the conceptual development and relevance of CTI-SOC2M2 and its structure, we conducted a number of expert interviews. The security experts selected for the interviews work in different industries and have different degrees

of knowledge about maturity models and SOC. While the role of two interviewees can be understood as senior SOC manager or analyst, the third interviewee has previously worked with maturity models. None of the participants is currently aware of a dedicated CTI-based SOC maturity model.

Design & Procedure: Using a semi-structured approach, the interview design and procedure includes the following four phases (Lazar et al., 2010):

1. **Introduction:** We start the interview to determine the interviewees understanding of common terminology and prior experiences with SOC services, CTI formats and maturity models. Afterwards, we introduce the proposed CTI-SOC2M2 and outline basic elements. We actively encourage interviewees to directly voice criticism and mention issues throughout the interview.
2. **CTI-SOC2M2:** In this phase we aim to get additional feedback on the individual SOC services and CTI formats. Although the result of an iterative process, we further aim to validate the capability maturity model with focus on its development and structure. We discuss methodological decisions with the interviewees to identify whether they support the relevance and applicability of the results. We also ask the participants to name aspects of CTI and SOC they see missing and which might require a more detailed explanation.
3. **Maturity Assessment:** This phase is focused on using the CTI-SOC2M2 and issues faced when doing so. To enable the interviewees to work with the proposed capability maturity model we explain aspects such as indicative question about CTI formats for capability levels and the foundations of maturity levels. During the interviews, the participants can also access the CTI-SOC2M2 tool. The primary goal in this interview phase is to identify whether the CTI-SOC2M2 approach is comprehensible, applicable and the self-assessment tool provides adequate functionality. We ask the interviewees whether there are further aspects they think would enhance the understanding and use of CTI-SOC2M2.
4. **Wrap-Up:** Last, we conclude the interviews with a summarizing discussion. We discuss with participants whether limitations to CTI-SOC2M2 are methodological, conceptual or based on implementation. This phase also includes collecting participants' ideas of features deemed useful and extensions to improve our approach.

Results: The interviews were scheduled for 60 minutes. The following results are first divided according to the four interview phases:

1. **Introduction:** Table 5 summarizes background information about the interviewed experts.
2. **CTI-SOC2M2:** Reflected by their knowledge the interviewees focused on CTI or SOC elements of the proposed CTI-SOC2M2. Above all, the interviewees unanimously stated the importance of an intelligence-driven SOC. The approach to assess SOC maturity with CTI and CTI formats was perceived as innovative. Paired with scientific methodology on CMM development the proposed model was seen as coherent. While the CTI formats were only

Table 5 – General information on the interview participants.

	Position	Business Branch	Organization's Size	SOC Knowledge	CMM Knowledge
#1	IT-Security Manager	Consulting	ca. 500.000	high	medium
#2	Senior Security Architect	Automotive	ca. 40.000	medium	medium
#3	Security Expert	Education	ca. 5.000	medium	high

partially known to the interviewees, the SOC services provided enough differentiation to cluster the essential SOC activities. As it was pointed out that SOC services are not independent, the participants referred to other organizational IT services. We see this as an important aspect and envision this within capability level *Integration* (e.g., organizational risk management). The completeness concern voiced by one interviewee had been addressed with a broad literature corpus used for taxonomy development. Concerning relevance and practical necessity to combine CTI and SOC, the participants all strongly agreed with a more data-centric approach in organizations.

3. *Maturity Assessment*: The participants' answers concerning the maturity assessment covered the step-wise approach to improving the maturity and further explanations about the capability levels. The mapping to the NIST incident response life cycle was seen as a helpful structuring element. One interviewee pointed out that the highest maturity level should emphasize the boldness of the CTI-focused SOC. Thus we re-considered our naming convention and opt for *visionary* as the highest maturity level. When explaining capability levels to the interviewees, it became apparent to direct future work to a more fine granular specification of data quality and possible metrics. The self-assessment tool was considered an essential element to the adoption of the CTI-SOC2M2, documenting the need for visualization.
4. *Wrap-Up*: The final phase revealed different perceptions on SOC and its services. Red teaming and threat hunting were topics of discussion as they apply only to sophisticated organizations with a strong focus on information security. In the same direction, the inclusion of active defense services beyond the use of honey pots was mentioned. We acknowledge that this is a possible SOC service. However, jurisdiction and existing laws in various regions prohibit its use. Therefore, we do not include this SOC service specifically.

We also apply the thematic synthesis by [Cruzes and Dybå \(2011\)](#) to the results of our expert interviews. Whereas the authors' approach is typically applied to academic literature and coding the content of primary studies, we use the interviews instead. Our starting point to thematic synthesis is the question: How do experts perceive CTI-SOC2M2? After the coding of data, the approach involves translating codes into themes. Within the expert interviews, we identified several codes. Due to the limited information available from the interviews, we identified a comparatively small number of codes. These codes were then directly transformed into four higher-order themes (see [Table 6](#)). Please note, each code constitutes a component of the respective theme. Also, we do not weigh and order the individual codes. Relevance, applicability, compre-

hensibility, and limitations represent the higher-order themes and build the model to answer the specified question.

Thematic synthesis of the expert interviews resulted in four higher-order themes – *relevance*, *applicability*, *comprehensibility*, and *limitations*. The following excerpts address some of the codes in [Table 6](#). Overall, the interviewees' feedback includes various aspects relating to the relevance of our proposed maturity model. As one interviewee stated, SOC services and CTI formats cannot be implemented separately but must be integrated (codes: SOC services, CTI concept). Further, a valuable contribution to practice stems from the CMM and the self-assessment tool. Here, interviewee perceptions include the flexibility of the model regarding specific CTI formats (code: flexibility). Comprehensibility is centered on the scientific methodology. The interviewees point to the conciseness of CTI-SOC2M2 based on CMM components (code: naming conventions). At last, the innovative approach faces limitations. This higher-order theme mainly concerns the granularity of the model (code: CTI data quality).

To subsume, experts perceive CTI-SOC2M2 as a relevant capability maturity model and see a valuable contribution. Applied to IT management, the self-assessment tool supports its actual use. In addition, the scientific methodology and the well-known reference framework foster comprehensibility. Nevertheless, some limitations must be accepted or addressed by the target audience itself.

[Cruzes and Dybå \(2011\)](#) conclude the thematic synthesis with an assessment of its trustworthiness. Trustworthiness is specified by the concepts of credibility, confirmability, dependability, and transferability. In the context of our thematic synthesis, credibility is addressed by the selection of the interviewees. We selected three interview participants that have sufficient experience in information security and are familiar with CMMs (see [Table 5](#)). Concerning confirmability, we opt for separating coding the interview results between different researchers and aggregating the outcome. Doing so allows us to avoid potential individual biases. Dependability as the stability of data is partially applicable to our synthesis. We assume the interview data to be stable. Finally, transferability refers to valuable insights beyond the scope of CTI formats, SOC and CMMs. The interview guide adapted for our purpose and the applied synthesis method document transferability.

8. Discussion

Novel aspects of this work are the integration of CTI and SOC services within CTI-SOC2M2. The capability maturity model provides an adequate foundation for assessing a SOC based on the CTI used. However, similar to other research efforts, this model has limitations, which are worth discussing. It should

Table 6 – Thematic synthesis of expert interviews: themes and codes.

Themes	Codes
Relevance	CMMs, SOC services, CTI concept, data-driven, CTI feeds, tactical level, strategic decisions, threat information, target audience
Applicability	Self-assessment, tool support, IT management, flexibility, existing CMMs, defined goal, SOC service selection, completeness
Comprehensibility	Naming conventions, scientific methodology, NIST incident response life cycle, stakeholders
Limitations	Active defense, CTI data quality, SOC service exclusion, metrics

be noted that the proposed model can only provide an indicator of overall SOC maturity, as the focus is exclusively on the integration of CTI and SOC services. Scoring a high maturity level in our CTI-focused model does not necessarily mean a high overall SOC maturity. Thus, a combination with more holistic models covering governance and roles might be recommended depending on the specific use case.

Furthermore, there is a challenge that applies to most maturity models, especially those developed in research. Both the development and the application always contain a certain degree of subjectivity, which can hardly be eliminated. In the development phase of the model, the degree of subjectivity can be controlled through methodology, but it cannot be avoided entirely. Also, when determining the capability levels using the model, a certain degree of subjectivity on the users' self-assessment cannot be avoided. In future work, the methodological procedure for developing the model could be supplemented by other methods. For example, conducting a Delphi study is a frequently chosen approach. However, from a research perspective, we decided to capture the current state-of-the-art presented in academic literature using a literature review and performing validation with expert interviews.

With the help of the two-stage evaluation, it could be shown that the problem definition is relevant and the proposed model appreciated by experts. However, many ways of maturity model evaluation could supplement the procedure described in this paper. We chose the most frequently used option conducting expert interviews, which is also the most useful for the present case.

Finally, it should be mentioned that for maturity models, completeness and perfection are strived for but not realized. Instead, a maturity model must be understood as a living model that evolves and is adapted to new requirements. The same applies to the present CTI-SOC2M2. Additional aspects such as CTI quality offer the opportunity to specify individual metrics but go beyond the scope of this paper. We are aware that weaknesses will emerge during practical use, which must then be addressed. In addition, the requirements for a SOC and the possibilities of CTI will change in the future, which is why the model must be expanded accordingly.

Implications for literature

As seen in Fig. 1, the CTI concept comprises application domains where CTI artifacts structured with CTI formats are used. This organizational CTI focus has not been examined by existing literature. Complementing existing research on individual CTI capabilities (Shin and Lowry, 2020), with our CTI-SOC2M2, we address the organizational CTI capabilities level, which is realized by mapping CTI formats to SOC services.

A second implication for literature is the aggregation of SOC literature concerning services and capabilities. While the literature corpus is based on previous work, we derive relevant information to structure SOC services for our model. In conjunction with CTI formats, two currently separate research areas are connected.

With CTI-SOC2M2, we build a first basis for the quest for mature, intelligence-driven security operations and incident response capabilities. However, for future work, we see the necessity to examine further 1) CTI quality and CTI automation, 2) incident response and CTI, and conduct 3) field studies based on CTI-SOC2M2.

Implications for industry

Maturity models evolved from best practices and have become popular in the industry. They are of particular interest at a higher management level. As some examples show, SOC-related models often provide a holistic view of SOC (see Table 2 and Section 5.1).

The popularity and use of CMMs in the industry are due to several reasons. One reason is that otherwise abstract factors, such as management success, can be measured with them. Regarding a SOC, it is possible to measure how it compares to other SOCs without consulting otherwise problematic metrics (e.g., the number of successful attacks). Another reason is that CMMs link theory and practice. More specifically, CMMs allow checking how close one's SOC comes to a theoretically and, in some cases, scientifically complete SOC.

In the previous presentation of the maturity model, we have taken a more academic and theoretical view. However, since a maturity model should be seen as a bridge between theory and practice, the implications of the model for the industry are presented subsequently.

In most cases, the current view of SOCs is people-driven and bases actions and decisions on the knowledge and intuition of analysts and other staff. Contrary, CTI-SOC2M2 aims at a more data-driven view, where the procedures within a SOC are based on available CTI – ideally resulting in a (partial) automation of incident handling processes. The use of CTI formats and SOC services in CTI-SOC2M2 contributes to security operations by guiding practitioners towards a more effective SOC.

Both SOC and CTI have only recently found their way into practice. However, different forms of SOC services and CTI formats predate their concepts. It is, therefore, necessary to align SOC services and CTI formats and adapt the latter to current needs and future requirements (e.g., Digital Twin based security operations (Dietz et al., 2020)). We contribute by emphasizing the importance of considering CTI and SOC together.

The resulting implications for industry can be subsumed as leveraging information about the external threat landscape. CTI formats lead to the operationalization of CTI in organizations and possibly improve proactive and reactive security measures. Organizations using CTI-SOC2M2 have a structured yet flexible model at hand to assess and improve their CTI-driven SOC maturity.

9. Conclusion

This paper presents CTI-SOC2M2, a capability maturity model that aims at assessing the maturity of SOCs based on the use of CTI. Special attention is paid to a structured and methodical approach. In addition to the maturity model itself, the contribution is divided into three parts: First, existing maturity models in the area of SOC, CTI, and incident response are collected and analyzed. Second, as a basis for the new maturity model, the activities in a SOC are clustered by services with the help of a structured literature review. Third, to finally develop the model, the most common CTI formats are mapped to the SOC services for which they are relevant. A mixed-method approach was performed to evaluate the relevance and applicability of the model. For this purpose, a quantitative user study and expert interviews were combined. The results show that the problem addressed by CTI-SOC2M2 is relevant and that the developed model is considered useful by experts. Implications resulting from CTI-SOC2M2 for literature center on the extension of existing research. The combined consideration of CTI and SOC leads to more mature security operations and incident response capabilities. Currently fragmented research is aggregated by our model. Implications for industry settle on the operationalization of CTI. Therefore, (semi)-structured CTI formats are leveraged to achieve implementation of the CTI concept in organizations. The proposed model is an essential step to assess the current state of a SOC and its ability to cope with external threat information. Its actual use within several organizations will eventually determine the models' success.

Credit Author Statement

DS and MV carried out the capability maturity model development and the underlying clustering of SOC services. DS contributed threat intelligence formats, mapping and assessment methodology while MV provided SOC content, mapping and implemented the prototype. GP participated in the formal definition of CTI-SOC2M2 supporting selection and structure of its elements. DS, MV and GP also helped to draft the manuscript revising it critically for important intellectual content. All authors read and approved the final manuscript.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This research was supported by the Federal Ministry of Education and Research (BMBF), Germany, as part of the DEVISE project and by the Bavarian Ministry of Economic Affairs, Regional Development and Energy (BayStMWi), as part of the IN-SIST project.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at [10.1016/j.cose.2021.102482](https://doi.org/10.1016/j.cose.2021.102482)

REFERENCES

- Ab Rahman NH, Choo K-KR. A survey of information security incident handling in the cloud. *computers & security* 2015;49:45–69.
- Acartürk C, Ulubay M, Erdur E. Continuous improvement on maturity and capability of security operation centres. *IET Inf. Secur.* 2020.
- Ahern DM, Clouse A, Turner R. *CMMI Distilled: A practical introduction to integrated process improvement*. Addison-Wesley Professional; 2004.
- Ahmad A, Hadgkiss J, Ruighaver AB. Incident response teams—challenges in supporting the organisational security function. *Computers & Security* 2012;31(5):643–52.
- Ahmad A, Maynard SB, Desouza KC, Kotsias J, Whitty MT, Baskerville RL. How can organizations develop situation awareness for incident response: a case study of management practice. *Computers & Security* 2021;101:102–22.
- Apache HTTP Server Project, 1995. NCSA Common Log Format. Last accessed 2021-07-01, <https://httpd.apache.org/docs/trunk/logs.html#common>.
- ArcSight, 2010. Common Event Format.
- ATC Project, 2020. RE&CT framework documentation. Last accessed 2021-02-01, <https://atc-project.github.io/atc-react/>.
- Baker DW, Christey SM, Hill WH, Mann DE. The development of a common enumeration of vulnerabilities and exposures, Vol. 7; 1999. p. 9.
- Bauer S, Fischer D, Sauerwein C, Latzel S, Stelzer D, Breu R. Towards an evaluation framework for threat intelligence sharing platforms. In: *Proceedings of the 53rd Hawaii International Conference on System Sciences*; 2020. p. 1–10.
- Becker J, Knackstedt R, Pöppelbuß J. Developing maturity models for it management. *Business & Information Systems Engineering* 2009;1(3):213–22.
- Bouwman X, Griffioen H, Egbers J, Doerr C, Klievink B, van Eeten M. A different cup of TI? the added value of commercial threat intelligence. In: *29th USENIX Security Symposium (USENIX Security 20)*; 2020. p. 433–50.
- Brown R, Lee RM. The evolution of cyber threat intelligence (cti): 2019 sans cti survey. SANS Institute 2019.
- Brown R, Lee RM. 2021 Sans cyber threat intelligence (cti) survey. SANS Institute 2021.
- Brown S, Gommers J, Serrano O. From cyber security information sharing to threat management. In: *Proceedings of the 2nd ACM workshop on information sharing and collaborative security*; 2015. p. 43–9.
- de Bruin T, Michael Rosemann, Ronald Freeze, Uday Kulkarni. Understanding the main phases of developing a maturity assessment model. *ACIS 2005 Proceedings - 16th Australasian Conference on Information Systems*, 2005.

- Caltagirone S, Pendergast A, Betz C. In: Technical Report. The diamond model of intrusion analysis. Center For Cyber Intelligence Analysis and Threat Research, Hanover Md; 2013.
- CAPEC Team, 2020. Schema documentation - schema version 3.4. Last accessed 2021-04-01, <https://capec.mitre.org/documents/schema/index.html>.
- Cheikes BA, Waltermire D, Scarfone K. In: Technical Report. Common Platform Enumeration: Naming Specification Version 2.3. Maryland, USA: National Institute of Standards and Technology; 2011. NIST Interagency Report 7695
- Chismon D, Ruks M. In: Technical Report. Threat intelligence: Collecting, analysing, evaluating. MWR InfoSecurity, CERT-UK; 2015.
- Christopher JD, Gonzalez D, White DW, Stevens J, Grundman J, Mehravari N, Dolan T. In: Technical Report. Cybersecurity Capability Maturity Model (C2M2). US Department of Energy (DOE); 2014.
- Cichonski P, Millar T, Grance T, Scarfone K. Computer security incident handling guide. NIST Special Publication 2012;800(61):1-147.
- CMMI Product Team, 2010. Cmmi for services, version 1.3: Improving processes for providing better services. https://resources.sei.cmu.edu/asset_files/TechnicalReport/2010_005_001_15290.pdf.
- CREST, 2014. Cyber Security Incident Response Maturity Assessment Tool (CSIR-MAT). <https://www.crest-approved.org/2018/07/20/cyber-security-incident-response-maturity-assessment/index.html>.
- CREST, 2016. Cyber Threat Intelligence Maturity Assessment Tool (CTI-MAT). <https://www.crest-approved.org/2020/01/10/cyber-threat-intelligence-maturity-assessment-tool/index.html>.
- Cruzes DS, Dybå T. Recommended steps for thematic synthesis in software engineering. In: 2011 international symposium on empirical software engineering and measurement. IEEE; 2011. p. 275-84.
- Dandurand L, Kaplan A, Kácha P, Kadobayashi Y, Kompanek A, Lima T, Millar T, Nazario J, Perlotto R, Young W. In: Technical Report. Standards and tools for exchange and processing of actionable information. European Union Agency for Network and Information Security (ENISA); 2014.
- Danyliw R. In: Technical Report. The Incident Object Description Exchange Format Version 2. Internet Engineering Task Force (IETF); 2016. <https://tools.ietf.org/html/rfc7970>.
- Dietz M, Vielberth M, Pernul G. Integrating digital twin security simulations in the security operations center. In: Proceedings of the 15th International Conference on Availability, Reliability and Security; 2020. p. 1-9.
- Dorling A. Spice: software process improvement and capability determination. *Software Quality Journal* 1993;2(4):209-24.
- Englbrecht L, Meier S, Pernul G. Towards a capability maturity model for digital forensic readiness. *Wireless Networks* 2020;26(7):4895-907.
- Farris KA, Shah A, Cybenko G, Ganesan R, Jajodia S. Vulcon: a system for vulnerability prioritization, mitigation, and management. *ACM Transactions on Privacy and Security* 2018;21(4):1-28. doi:10.1145/3196884.
- Forum of Incident Response and Security Teams (FIRST), 2019. Common Vulnerability Scoring System version 3.1: Specification document - revision 1. Last accessed 2021-02-01, <https://www.first.org/cvss/specification-document>.
- Gerhards R, et al. In: Technical Report. The syslog protocol. RFC 5424, March; 2009.
- Hámorník BP, Krasznay C. A Team-level Perspective of Human Factors in Cyber Security: Security Operations Centers. In: *Advances in Intelligent Systems and Computing*, Vol 593. Cham: Springer International Publishing; 2018. p. 224-36. doi:10.1007/978-3-319-60585-2_21.
- Hernandez-Ardieta JL, Tapiador JE, Suarez-Tangil G. Information sharing models for cooperative cyber defence. In: 2013 5th International Conference on Cyber Conflict (CYCON 2013). IEEE; 2013. p. 1-28.
- Humphrey WS. Characterizing the software process: a maturity framework. *IEEE Software* 1988;5(2):73-9. doi:10.1109/52.2014.
- Hutchins EM, Cloppert MJ, Amin RM. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. *Leading Issues in Information Warfare & Security Research* 2011;1(1):80.
- Islam C, Babar MA, Nepal S. A multi-vocal review of security orchestration. *ACM Computing Surveys (CSUR)* 2019;52(2):1-45.
- Jacobs P, Arnab A, Irwin B. In: 2013 Information Security for South Africa. Classification of security operation centers. IEEE; 2013. doi:10.1109/ISSA.2013.6641054.
- Kokulu FB, Soneji A, Bao T, Shoshitaishvili Y, Zhao Z, Doupé A, Ahn G-J. Matched and mismatched socs: A qualitative study on security operations center issues. In: Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security; 2019. p. 1955-70.
- Kowtha S, Nolan LA, Daley RA. Cyber security operations center characterization model and analysis. In: 2012 IEEE Conference on Technologies for Homeland Security (HST). IEEE; 2012. p. 470-5.
- Krebs, B., 2021. A basic timeline of the exchange mass-hack. <https://krebsonsecurity.com/2021/03/a-basic-timeline-of-the-exchange-mass-hack/>.
- Lakshmi, R, Naseer, H, Maynard, S, Ahmad, A. Sensemaking in cybersecurity incident response: The interplay of organizations, technology and individuals. arXiv preprint arXiv:2107.02941 2021.
- Lazar J, Feng JH, Hochheiser H. Research methods in human-Computer interaction. Burlington: Morgan Kaufmann; 2010.
- Li VG, Dunn M, Pearce P, McCoy D, Voelker GM, Savage S. Reading the tea leaves: A comparative analysis of threat intelligence. In: 28th USENIX Security Symposium (USENIX Security 19); 2019. p. 851-67.
- Lourenco M. In: Technical Report. CTI Capability Maturity Model. European Union Agency for Network and Information Security (ENISA); 2018.
- Luchs M, Doerr C. In: Technical Report. Measuring your Cyber Threat Intelligence Maturity. Hasso Plattner Institut and TU Delft; 2020.
- Madani A, Rezayi S, Gharaee H. Log management comprehensive architecture in security operation center (soc). In: 2011 International Conference on Computational Aspects of Social Networks (CASoN). IEEE; 2011. p. 284-9. doi:10.1109/CASON.2011.6085959.
- Mavroeidis V, Bromander S. Cyber threat intelligence model: an evaluation of taxonomies, sharing standards, and ontologies within cyber threat intelligence. In: 2017 European Intelligence and Security Informatics Conference (EISIC). IEEE; 2017. p. 91-8.
- Menges F, Pernul G. A comparative analysis of incident reporting formats. *Computers & Security* 2018;73:87-101. doi:10.1016/j.cose.2017.10.009.
- Mettler, T., 2009. A design science research perspective on maturity models in information systems.
- Microsoft, 2018. Windows Event Log. Last accessed 2021-07-01, <https://docs.microsoft.com/en-us/windows/win32/wes/windows-event-log>.
- Microsoft Threat Intelligence Center (MSTIC). In: Technical Report. HAFNIUM targeting Exchange Servers with 0-day

- exploits. Microsoft; 2021. <https://www.microsoft.com/security/blog/2021/03/02/hafnium-targeting-exchange-servers/>.
- MITRE, 2020. Common Weakness Enumeration - a community-developed list of software & hardware weakness types. Last accessed 2021-02-01, <https://cwe.mitre.org/index.html>.
- Neiva C, Lawson C, Bussa T, Sadowski G. In: Technical Report. 2020 Market Guide for Security Orchestration, Automation and Response Solutions. Gartner; 2020.
- Nickerson RC, Varshney U, Muntermann J. A method for taxonomy development and its application in information systems. *European Journal of Information Systems* 2013;22(3):336–59. doi:10.1057/ejis.2012.26.
- OASIS. Open command and control (openc2) language specification version 1.0 - Committee specification 02. OASIS; 2020. Last accessed 2020-11-15, <https://docs.oasis-open.org/openc2/oc2ls/v1.0/cs02/oc2ls-v1.0-cs02.html>.
- OASIS. CACAO Security playbooks version 1.0 - Committee specification 01. OASIS; 2021. Last accessed 2021-01-15, <https://docs.oasis-open.org/cacao/security-playbooks/v1.0/security-playbooks-v1.0.html>.
- OASIS Cyber Threat Intelligence (CTI) Technical Committee. STIX™ Version 2.1: Committee specification 01. OASIS; 2020. Last accessed 2021-01-01, <https://docs.oasis-open.org/cti/stix/v2.1/stix-v2.1.html>.
- OASIS Cyber Threat Intelligence (CTI) Technical Committee. TAXII™ Version 2.1: Committee specification 01. OASIS; 2020. Last accessed 2020-10-20, <https://docs.oasis-open.org/cti/taxii/v2.1/taxii-v2.1.html>.
- Onwubiko C. Cyber security operations centre: Security monitoring for protecting business and supporting cyber defense strategy. In: 2015 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA). IEEE; 2015. p. 1–10. doi:10.1109/CyberSA.2015.7166125.
- Onwubiko C, Ouazzane K. Soter: a playbook for cybersecurity incident management. *IEEE Trans. Eng. Manage.* 2020;1–21. doi:10.1109/TEM.2020.2979832.
- RawSec - Quentin Jerome, 2018. Go EvtX Signature Engine. Last accessed 2021-04-01, <https://rawsec.lu/doc/gene/1.6/writerules.html>.
- Rea-Guaman AM, San Feliu T, Calvo-Manzano JA, Sanchez-Garcia ID. Comparative study of cybersecurity capability maturity models. In: *International Conference on Software Process Improvement and Capability Determination*. Springer; 2017. p. 100–13.
- Ross, D., Shiffer, J., Dell, T., Gibb, W., Wilson, D., 2013. OpenIOC 1.1 Schema. Last accessed 2021-04-01, https://github.com/mandiant/OpenIOC_1.1.
- Roth, F., Patzke, T., 2017. Sigma - Generic Signature Format for SIEM Systems. Last accessed 2021-04-01, <https://github.com/SigmaHQ/sigma/wiki/Specification>.
- Schaberreiter T, Kupfersberger V, Rantos K, Spyros A, Papanikolaou A, Ilioudis C, Quirchmayr G. A quantitative evaluation of trust in the quality of cyber threat intelligence sources. In: *Proceedings of the 14th International Conference on Availability, Reliability and Security*; 2019. p. 1–10.
- Schinagl S, Schoon K, Paans R. A framework for designing a security operations centre (soc). In: *2015 48th Hawaii International Conference on System Sciences*. IEEE; 2015. p. 2253–62.
- Schlette D, Böhm F, Caselli M, Pernul G. Measuring and visualizing cyber threat intelligence quality. *Int. J. Inf. Secur.* 2021;20(1):21–38.
- Settanni G, Shovgenya Y, Skopik F, Graf R, Wurzenberger M, Fiedler R. Acquiring cyber threat intelligence through security information correlation. In: *2017 3rd IEEE International Conference on Cybernetics (CYBCONF)*. IEEE; 2017. p. 1–7. doi:10.1109/CYBCONF.2017.7985754.
- Shah A, Ganesan R, Jajodia S. A methodology for ensuring fair allocation of csoc effort for alert investigation. *Int. J. Inf. Secur.* 2019;18(2):199–218. doi:10.1007/s10207-018-0407-3.
- Shin B, Lowry PB. A review and theoretical explanation of the 'cyberthreat-intelligence (cti) capability' that needs to be fostered in information security practitioners and how this can be accomplished. *Computers & Security* 2020;92:101761.
- Sillaber C, Sauerwein C, Mussmann A, Breu R. Towards a maturity model for inter-organizational cyber threat intelligence sharing: a case study of stakeholders' expectations and willingness to share. *Proceedings of Multikonferenz Wirtschaftsinformatik (MKWI 2018)* 2018:1409–20.
- Skopik F, Settanni G, Fiedler R. A problem shared is a problem halved: a survey on the dimensions of collective cyber defense through security information sharing. *Computers & Security* 2016;60:154–76. doi:10.1016/j.cose.2016.04.003.
- Snort Team, 2021. Writing Snort Rules. Last accessed 2021-04-01, <https://www.snort.org/documents>.
- Stikvoort D. In: Technical Report. SIM3: Security Incident Management Maturity Model. OCF, S-CURE and PRESECURE; 2015.
- Strom BE, Applebaum A, Miller DP, Nickels KC, Pennington AG, Thomas CB. In: Technical Report. MITRE ATT&CK: Design and philosophy. The MITRE Corporation; 2018.
- Taurins E. In: Technical Report. How to set up CSIRT and SOC - Good Practice Guide. European Union Agency for Network and Information Security (ENISA); 2020.
- The Zeek Project, 2021. Signature Framework. Last accessed 2021-04-01, <https://docs.zeek.org/en/current/frameworks/signatures.html>.
- Tounsi W, Rais H. A survey on technical threat intelligence in the age of sophisticated cyber attacks. *Computers & security* 2018;72:212–33.
- Van Os, R., 2016. SOC-CMM: Designing and evaluating a tool for measurement of capability maturity in security operations centers.
- VERIS Community, 2021. Veris - the vocabulary for event recording and incident sharing. Last accessed 2021-04-01, <http://veriscommunity.net/index.html>.
- Vielberth M, Böhm F, Fichtinger I, Pernul G. Security operations center: a systematic study and open challenges. *IEEE Access* 2020;8:227756–79.
- VirusTotal - Victor Alvarez, 2014. Signature Framework. Last accessed 2021-04-01, <https://yara.readthedocs.io/en/stable/>.
- Wagner C, Dulaunoy A, Wagener G, Iklody A. MISP - the design and implementation of a collaborative threat intelligence sharing platform. In: Katzenbeisser S, Weippl E, Blass E-O, Kerschbaum F, editors. In: *Proceedings of the 2016 ACM on Workshop on Information Sharing and Collaborative Security - WISCS'16*. ACM; 2016. p. 49–56. doi:10.1145/2994539.2994542.
- Zimmerman C. In: Technical Report. Cybersecurity Operations Center. The MITRE Corporation; 2014.
- Daniel Schlette** received the master's degree (Hons.) in management information systems from the Elite Graduate Program, University of Regensburg, in 2019. He is currently pursuing the Ph.D. degree with the Chair of Information Systems, University of Regensburg. Since 2019, he has been a Research Assistant with the Chair of Information Systems, University of Regensburg. His research interests include the field of Cyber Threat Intelligence. His primary focus within this topic is to leverage structured data formats, explore aspects of Security Orchestration, Automation and Response (SOAR), and data quality.
- Manfred Vielberth** received the bachelor's and master's degrees in management information systems with a specialization in cyber security from the University of Regensburg, Germany. Since February 2017, he is pursuing the Ph.D. degree with the Chair of Information Systems, University of Regensburg. His research interest

includes human aspects in the security analytics domain. On the expert side, this mainly comprises improving processes for better integrating security analysts within a Security Operations Center. In terms of security novices, this primarily covers capturing reports about security incidents in the context of the Human-as-a-Security-Sensor paradigm.

Günther Pernul (Member, IEEE) received the diploma and Ph.D. degrees (Hons.) in business informatics from the University of Vienna, Austria. He is currently a Professor with the Department

of Information Systems, University of Regensburg, Germany. Previously, he held positions at the University of Duisburg-Essen, Germany; the University of Vienna; the University of Florida, Gainesville; and the College of Computing, Georgia Institute of Technology, Atlanta. His research interests include data and information-security aspects, data protection and privacy, data analytics, and advanced datacentric applications.

4 SOAR4IoT: Securing IoT Assets with Digital Twins

Publication information

Current status:	Published
Conference:	17th International Conference on Availability, Reliability and Security, Vienna, Austria, August 23 - 26, 2022
Date of acceptance:	02 March 2022
Full citation:	EMPL, P., SCHLETTE, D., ZUPFER, D., & PERNUL, G. (2022). SOAR4IoT: Securing IoT Assets with Digital Twins. In <i>The 17th International Conference on Availability, Reliability and Security (ARES 2022)</i> , pp. 4:1-4:10.
Authors' contributions:	Philip Empl 40% Daniel Schlette 40% Daniel Zupfer 10% Günther Pernul 10%

Conference description: The International Conference on Availability, Reliability and Security (ARES) brings together researchers and practitioners in the field of IT security & privacy. Since 2005, ARES serves as an important platform to exchange, discuss and transfer knowledge and is hosted every year in another European city.



SOAR4IoT: Securing IoT Assets with Digital Twins

Philip Empl*
philip.empl@ur.de
University of Regensburg
Germany

Daniel Schlette
daniel.schlette@ur.de
University of Regensburg
Germany

Daniel Zupfer
daniel.zupfer@ur.de
University of Regensburg
Germany

Günther Pernul
guenther.pernul@ur.de
University of Regensburg
Germany

ABSTRACT

As more and more security tools provide organizations with cybersecurity capabilities, security analysts are overwhelmed by security events. Resolving these events is challenging due to extensive manual processes, limited financial resources, and human errors. Security Orchestration, Automation, and Response (SOAR) is an established approach to manage security tools and assets. However, SOAR platforms typically integrate traditional IT systems only. Additional considerations are required to deal with the Internet of Things (IoT), its multiple devices and complex networks. Therefore, we adapt SOAR to IoT. We first aggregate existing research and information on SOAR and SOAR platforms. We envision the SOAR4IoT framework, making IoT assets manageable for SOAR via middleware. We implement a prototypical digital twin-based SOAR application integrating IoT assets and security tools to validate our framework. The experimental setup includes two playbooks coping with Mirai and Sybil attacks. Results show feasibility as our SOAR application enables securing IoT assets with digital twins.

CCS CONCEPTS

• **Security and privacy** → *Network security; Systems security; Security services*; • **Computer systems organization**; • **Information systems**;

KEYWORDS

Internet of Things, Security Orchestration, Incident Response, SOAR, Digital Twin

ACM Reference Format:

Philip Empl, Daniel Schlette, Daniel Zupfer, and Günther Pernul. 2022. SOAR4IoT: Securing IoT Assets with Digital Twins. In *The 17th International Conference on Availability, Reliability and Security (ARES 2022)*, August 23–26, 2022, Vienna, Austria. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3538969.3538975>

1 INTRODUCTION

Attackers and defenders shape cybersecurity. Sophisticated attacks on networked information systems are countered by defenders' use of tools for security monitoring and operations. However, there is an ongoing challenge for security analysts. While more and more

security tools are being used, analysts can face up to 11,000 security alerts per day (including false positives) [11]. Therefore, organizations use Security Orchestration, Automation and Response (SOAR) platforms promising tool integration, automation, and streamlined workflows for rapid incident response [19, 25].

SOAR platforms are based on security events. Security events concern traditional IT resources but also the Internet of Things (IoT). The new IoT frontier (e.g., smart factories or automated home systems) with its multitude of heterogeneous devices contributes to the ongoing datafication but currently neglects cybersecurity. Inadequate or missing security measures caused by a “set-it-and-forget-it manner” [20] are illustrative for the insecurity of IoT assets. Attackers notice these IoT security issues, as Kaspersky reports 1.5 billion attacks against their IoT honeypots in the first half of 2021 [30]. Eventually, networked IoT devices exposing default username/password authentication will become part of botnets. Estimates see the approximate time to compromise an IoT device at just five minutes [20]. Thus, it is necessary to extend security operations to IoT assets for which digital twins provide promising features [9]. Digital twins are used for security to simulate IoT attacks [8] and can assist incident response [7, 10].

Whether IoT-specific or not, security analysts cannot process security events manually. SOAR platforms greatly help analysts perform investigations and initiate adequate incident response actions. Analysts can reduce time and resources spent on low-priority events and manual actions using automated playbooks. Thus, SOAR documents a shift towards more effective security operations within organizations. As SOAR attracts attention in research and provides the dynamics to abstract complex environments, we investigate its potential for the IoT. Consequently, we ask “*how to use Security Orchestration, Automation and Response for the Internet of Things?*” We expect the general applicability of SOAR for IoT as it is a flexible construct. Still, it is crucial to showcase adaptation rigorously.

In this paper, we aim to answer the following questions: (1) What defines SOAR? (2) How to secure the IoT? (3) How to implement SOAR for IoT with digital twins? These questions lead to our main contributions:

- We enlighten SOAR core activities and platform features by analyzing the few academic works and current SOAR platforms.
- We envision our SOAR4IoT framework built on IoT attacks and mitigation strategies. Our framework encompasses IoT assets, middleware, SOAR platform, and security tools.
- We provide a SOAR4IoT implementation leveraging digital twins. The experimental setup documents the straightforward, ground-up implementation of a SOAR platform, including Eclipse Ditto-based digital twins, which researchers and practitioners can easily adapt and extend.

*Corresponding author



This work is licensed under a Creative Commons Attribution International 4.0 License.

ARES 2022, August 23–26, 2022, Vienna, Austria
© 2022 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-9670-7/22/08.
<https://doi.org/10.1145/3538969.3538975>

- We explore two security issues of IoT assets. We address IoT security operations by designing and implementing two generic playbooks for orchestration and automated response to the Mirai botnet and the Sybil attack.

The paper is organized as follows. Section 2 outlines IoT, digital twins for cybersecurity, SOAR foundations and describes related work. Section 3 elaborates the framework defining the characteristics of SOAR, discussing the objectives of secure IoT assets, and describing technologies abstracting the IoT. Then, formal requirements lead to the overall SOAR4IoT framework. We validate our framework in Section 4 through the implementation of a digital twin-based SOAR platform integrating two use cases. We conclude our paper in Section 5.

2 BACKGROUND AND RELATED WORK

This section elaborates the background on IoT (Section 2.1), digital twins for cybersecurity (Section 2.2) and SOAR (Section 2.3), concluding with related work (Section 2.4).

2.1 Internet of Things

The IoT is characterized by identifiable networking objects (sensors or actuators) advertising their services to assemble semantic-rich applications [1]. Beyond scrutinizing particular devices, the IoT involves communication, applications, and processes. Heterogeneous devices and machines of widely ranging specifications and data operate seamlessly and collaboratively to assist business processes. The heterogeneity of IoT devices and networks is mainly caused by various manufacturers and (communication) protocols. As a result, there are plenty of cybersecurity issues demanding 1) automated security operations (detection and mitigation) and 2) orchestration of security functions for the IoT [17]. When it comes to integrating IoT assets, middleware is reliable, and a common choice [27, 32]. Organizations can choose between different types of middleware according to technology preferences and use cases (see Figure 1).

2.2 Digital Twins for Cybersecurity

In general, digital twins can be conceived as middleware. At its core, the digital twin links a virtual representation to a physical asset aiming to mirror the asset along its life cycle with semantic technologies [3]. The digital twin synchronizes system states using bidirectional communication with its physical counterpart. Implementing digital twins is a challenging task. Digital twins (e.g., Eclipse Ditto or Azure Digital Twins) can be used standalone or connected to IoT platforms (e.g., Eclipse Kapua or Azure IoT Hub).

From a security perspective, digital twins concern three primary security-operation modes: replication, simulation, and historical data analytics [8]. *Historical data analytics* deals with the documented behavior of IoT assets in the past and draws conclusions for the future. *Simulations* build on user-specific parameters and model the semantics of the real world. Last, the *replication* integrates real-world data to semantically model and operate a digital twin identical to its real-world counterpart. These operation modes assist security operations. For instance, behavior-based modeling supports more efficient intrusion detection, and the virtual representation of the digital twin is suitable for security training [9]. Moreover,

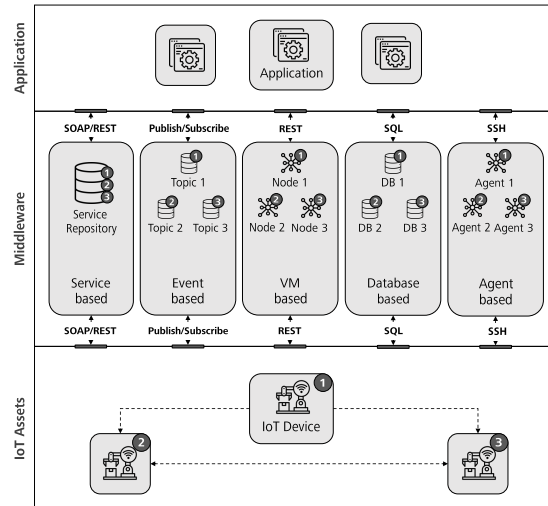


Figure 1: IoT architecture and middleware types

replication-based digital twins indicate security orchestration and incident response features.

2.3 Security Orchestration, Automation and Response (SOAR)

Platforms promising *Security Orchestration, Automation and Response (SOAR)* capabilities for organizations are the latest solutions proposed by cybersecurity vendors [19]. Like other solutions before, the underlying concept has not received much research attention while products are being pushed to market. SOAR is not a standalone concept but part of continuous development. From related concepts like log management to Security Information and Event Management (SIEM), Cyber Threat Intelligence (CTI), and security orchestration, it can be observed that succeeding concepts build on previous ones. Examining SOAR, it becomes evident that platforms, system architectures, and data are crucial to understanding and implementing the concept.

In the organizational context, SOAR and corresponding platforms are associated with the Security Operations Center (SOC) or Computer Security Incident Response Team (CSIRT) [31]. Intuitively, SOAR aims to assist activities within the three domains of 1) security orchestration, 2) automation, and 3) incident response.

For *security orchestration*, SOAR subsumes the functionality of SIEM and integrates multiple devices, systems, and security tools [13]. Additionally, integration and unification aspects of SOAR relate to threat intelligence as relevant information about threats, attacks, and vulnerabilities is aggregated from internal and external sources. For *automation*, SOAR relies on events and defined courses of action to enable rapid security operations. Thus, automation bridges the gap between security orchestration and incident response. For *incident response*, containment, eradication, and recovery activities demand to derive and perform appropriate measures.

Therefore, SOAR includes the instrumentalization of endpoints and security tools to execute commands.

Related to SOAR is the standardization and representation of incident response [28]. While current systems are often based on ticketing systems for security incidents, incident response playbooks are central. In essence, incident response playbooks define how to conduct a specified defensive procedure. Towards standardization, the incident response community initiated the development of dedicated data formats. These formats specify structural elements and required meta-data for incident response use cases. For instance, the two formats *Open Command and Control (OpenC2)* [23] and *Collaborative Automated Course of Action Operations (CACAO) for Cyber Security* [24] document different focal areas such as executable commands and procedural workflows, respectively.

2.4 Related Work

IoT devices and networks are susceptible to cyberattacks. Providing security measures for IoT is a practical problem and has attracted researchers' attention. As outdated firmware enables attacks on IoT devices, the literature emphasizes security orchestration by using a firmware update scenario (e.g., [2]). RFC 9019 describes updating IoT firmware in detail [18] while others use distributed ledger technologies [5]. As a consequence, we consider IoT firmware updates to validate our work. From a network perspective, the European Telecommunication Standard Institute proposes central security orchestration based on automated configurations and deployments [15]. We build on existing research and unify security orchestration activities. We include network and device layers within a single SOAR framework.

Digital twins for incident response is a trending research topic. Digital twins assist analysts in SOC [8] and are proposed for response measures [12]. Especially for operational systems, digital twins should implement cybersecurity services (e.g., access control, intrusion detection, or incident response) [7]. In a recent publication, Eckhart and Ekelhart [10] emphasize digital twins of real-world IoT systems as a new method for incident response. Existing literature only conceptualizes digital twin-based incident response. We are taking research further and implement digital twins for incident response.

Scoping the topic of SOAR, we identified additional related work. Most notably, Islam et al. [13] provide a survey on security orchestration. In a follow-up work on SOAR architecture, the authors propose the layered integration of security tools and map tools to response activities [14]. For CTI sources in SOAR, security enumerations have been discussed in the context of the IoT [29]. We go beyond security tools and include application aspects and IoT assets in our approach.

Further, SOAR has been examined in the context of incident response. Complementary to incident response formats, Schlette et al. [28] outline the vast SOAR product landscape. As SOAR platforms assist organizations' incident response, research addressed the appropriate selection [22] and quantitative evaluation of features [21]. SOAR platforms evolve and existing works provide a snapshot. Based on these works, we aggregate common features of SOAR platforms and settle on agreed-upon characteristics.

Table 1: SOAR requirements

	Requirement	Description	IoT
Core activities	Security Orchestration	Integration of IT assets, security tools, and threat intelligence	*
	Automation	Use of technologies and logic to perform security operations	✓
	Incident Response	Investigation, mitigation, and remediation of incidents	*
Platform features	User Interface	Dashboard or console for human interaction	✓
	Playbooks	Workflows, courses of action, or scripts	✓
	Ticketing System	Case management for security incidents	✓
	User Management	Access control and communication	✓
	✓ is applicable	* requires modification	

3 SOAR4IOT FRAMEWORK

To apply SOAR to IoT, we first identify general SOAR requirements (Section 3.1). Examining attacks on the IoT, we then derive IoT incident response objectives (Section 3.2). These objectives guide us towards required IoT security orchestration (Section 3.3). Based on our formal model (Section 3.4), we conceptualize a SOAR4IoT framework (Section 3.5) that integrates IoT systems using digital twins.

3.1 SOAR Requirements

SOAR requirements describe essential characteristics for the implementation of SOAR. Ultimately, SOAR requirements can assist the development of a SOAR platform, the evaluation of existing ones, or the adaptation to IoT devices and networks. In the following, we aggregate SOAR requirements from existing literature and validate the findings by examining current SOAR platforms. Table 1 describes core activities and platform features.

Core activities (i.e., security orchestration, automation, and incident response) constitute one group of requirements. They represent platform capabilities. For IoT, security orchestration demands modification as heterogeneous, dispersed devices form dynamic networks. Task automation remains largely unaffected, is conducted at SOAR platform level, and applies to IoT. Incident response measures directly involve IoT assets and thus demand modification.

Platform features constitute the second group of SOAR requirements. They represent technical aspects of a SOAR platform. Typically, a SOAR platform provides a user interface such as a dashboard or a console to assist orchestration and response activities [14]. More precisely, the user interface allows querying data and triggering courses of action. Playbooks are another dedicated SOAR

platform feature [21]. Playbooks represent workflows including actuators, actions, and artifacts to support automation and incident response. For instance, a remediation playbook can be designed and configured to make an orchestrated device (i.e., actuator) install (i.e., action) a new firmware version (i.e., artifact). Linked to security incidents or threat intelligence, (semi-)automation is possible. A ticketing system is a SOAR platform feature that helps to keep track of security incidents [13]. Tickets and case management also support prioritization and relate to security events. At last, SOAR platforms enable collaboration and include user management [22]. The platform-centric features above apply to SOAR for IoT.

Aside from literature and their analysis, we also analyzed a selected few SOAR platforms (Cortex XSOAR, D3 XGEN SOAR, Simplify, Splunk SOAR, Tines). In addition, the latest Gartner market report [19] reveals some information on SOAR requirements. Our observations of SOAR platform characteristics include:

- Ready-to-use connectors, adapters, or similar interfaces
- No-code or low-code approach for playbooks
- SIEM functions included or integrated
- Ticketing system included or integrated

Most notably, SOAR platforms acknowledge the multitude of other security tools and provide necessary technical integrations. Playbook editors emphasize visualization and drag-and-drop functionality but also allow to generate scripts. Concerning SIEM functions, we consider log collection, detection, correlation, and alerts to be SIEM characteristics. However, some SOAR platforms directly include these functions. Moreover, there is only an arbitrary boundary between some SIEM and SOAR tools (e.g., Wazuh). Ticketing systems build an underlying foundation for SOAR platforms and are closely related to correlation and prioritization. Nevertheless, organizations can also integrate existing security ticketing systems.

As a result, core activities and platform features apply to current SOAR platforms. In the context of IoT and our framework, SOAR requirements are applicable but also demand adaptation.

3.2 IoT Incident Response Objectives

We discuss possible attacks and vulnerabilities of IoT systems to identify relevant assets that necessitate SOAR. The IoT provides a favored attack surface to different threat actors pooling their resources. As the number of IoT market participants grows, time-to-market is shortening, standards are lacking, and security is affected. Consequently, inadequate security of IoT assets is a call to incident response (e.g., update procedures or configuration). Research identifies several perspectives on IoT attacks and vulnerabilities, such as encryption attacks [16], attacks mapped to the ISO/OSI stack [4], or the most common vulnerabilities listed by OWASP IoT Top 10¹. We distinguish IoT attacks on a higher level. Thereby we differentiate between attacks on device-level and network-level. We exclude attacks concerning other layers than the physical or network layer (e.g., attacks in cloud environments) because these attacks are not unique to the IoT. In summary, IoT attacks target:

- Device-level – hardware-based attacks, software-based attacks, and sensor data-based attacks
- Network-level – network-based attacks

¹<https://owasp.org>

Table 2: IoT attacks and mitigations

Type	Attack	Mitigation
Hardware-based	Node tampering	Perimeter security
Software-based	Mirai malware	Firmware update
Data-based	False injection	Authentication
Network-based	Sybil attack	Offboarding

Hardware-based attacks target the physical layer to damage IoT devices systematically. These physical layer attacks include node tampering, node jamming, or other physical damage. Software-based attacks on IoT devices usually involve firmware vulnerabilities or the (embedded) operating system. These vulnerabilities are exploited by well-known malware, such as Mirai botnet, Industroyer, or Reaper. Attacks also target data, especially sensor data. Injecting false data, eavesdropping and task inference are data-based attacks and conclude the device-level attacks. The network-level scopes all attacks based on the ISO/OSI stack layers, e.g., Sybil attack, denial of service, or wormhole attack.

In order to mitigate and prevent these vulnerabilities and attacks, security measures concerning IoT are discussed [4]. These security measures constitute IoT incident response objectives. More generally, there are proactive and reactive security measures. For instance, over-the-air (OTA) firmware updates and strengthening of password security are proactive security measures and the on- and offboarding of IoT devices count to reactive security measures. SOAR platforms can orchestrate proactive and reactive security measures. We do not consider security-by-design decisions (e.g., encryption mechanisms).

The orchestration of IoT devices and networks is a prerequisite to incident response. Playbooks are a crucial platform feature of SOAR to enable automation. Referring back to SOAR requirements, the deployment of the other two core capabilities, namely orchestration and response, in the IoT is challenging. While the orchestration of security tools is similar to traditional SOAR and requires no further considerations, the orchestration of IoT devices and networks requires more attention. Table 2 summarizes attacks on IoT assets and example mitigations. Moreover, different means of IoT security orchestration exist, which we identify in the next section.

3.3 IoT Security Orchestration

IoT security orchestration is directed at *IoT devices* (device-level) and *IoT networks* (network-level). Security measures for hardware-based attacks are enabled by manual tasks only. Proactively locking IoT devices away is an illustrative physical security measure and not part of SOAR.

In general, middleware is used to abstract IoT devices and their functionalities. However, middleware can also serve security orchestration purposes. Commercial solutions address IoT devices with two common middleware concepts: Digital twins and IoT platforms. Our work takes on a broad perspective but emphasizes the digital twin concept for representing IoT assets.

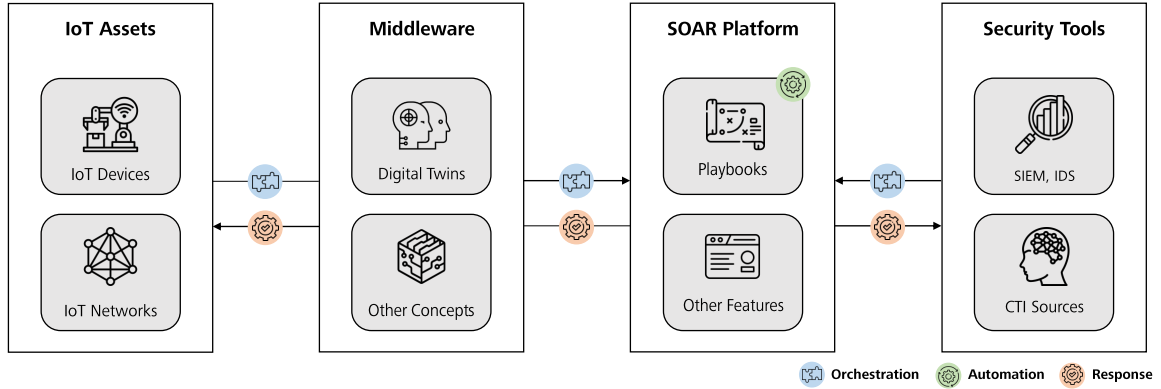


Figure 2: SOAR4IoT framework

Digital twins provide many features that enable security orchestration for IoT devices. They go beyond IoT platforms that are centered on common management tasks (e.g., onboard, monitor, and offboard devices). In particular, digital twins in replication mode provide IoT device modeling and security features. The bidirectional communication between the digital twin and IoT asset is beneficial as synchronizing sensor data and receiving commands can fulfill security orchestration. For instance, digital twins can store threat information acquired from third-party apps and synchronize information about vulnerabilities with their physical counterparts.

Besides IoT devices, digital twins and IoT platforms also extend to IoT networks. In this regard, digital twins allow the representation of dedicated edge nodes. Edge nodes are used in IoT networks as they control device communication. Using digital twins of edge nodes is thus a node-centric approach to communication-related security orchestration.

To sum things up, IoT device orchestration is enabled by digital twins. Further, IoT network orchestration requires the integration of edge nodes. Edge nodes are crucial as they control sub-networks containing several IoT devices. Therefore, we also include some node-centric aspects of IoT networks in our framework. We consider edge nodes represented by digital twins.

3.4 Formal Model

Concerning the security objectives of IoT, we define requirements targeting the three core capabilities of SOAR. These requirements are essential for the implementation of SOAR platforms and the definition of playbooks. The formal model includes:

REQUIREMENT 1 (ORCHESTRATION OF IoT ASSETS). We denote IoT assets as $A = \{a_1, a_2, \dots, a_n\}$, whereby an asset is either a device, network or security tool. These assets are integrated into *SOAR*:

$$a \rightarrow \text{SOAR}$$

REQUIREMENT 2 (AUTOMATION OF SECURITY MEASURES). Automation depends on security measures strategically executed for a specific event $E = \{e_1, e_2, \dots, e_o\}$ mapping an asset to a playbook $P = \{p_1, p_2, \dots, p_m\}$. Thereby, a playbook is generic and could be

linked to one or more assets, located inside a SOAR platform. An asset does not necessarily require a playbook:

$$\exists e \in E : e \rightarrow \text{SOAR}(p \circ a) \wedge \text{SOAR}(p) \rightarrow a$$

$$\exists a \in A : \neg \text{SOAR}(p \circ a)$$

REQUIREMENT 3 (DEPLOYMENT OF RESPONSES TO IoT ASSETS). Response of the SOAR platform depends on whether at least one playbook fulfills or characterizes appropriate security measures for an event. Otherwise, no response is automatically deployed:

$$\text{respond}(e) = \begin{cases} \text{SOAR}(p) \rightarrow a & \text{if } \exists p \in P : \text{SOAR}(p \circ e) \\ \text{notify}(e) & \text{otherwise.} \end{cases}$$

In the next step, we outline our framework, its components and middleware integration.

3.5 Framework Overview

Middleware integration complements our SOAR4IoT framework. We emphasize using digital twin middleware to extend existing SOAR platforms based on the previously established SOAR requirements and IoT security objectives. Figure 2 depicts the SOAR4IoT framework and the middleware integration.

IoT assets. The SOAR4IoT framework is based on IoT assets. IoT assets are classified as IoT devices (i.e., sensors or actuators) or IoT networks (i.e., edge nodes and communication). Intertwined, IoT devices and networks form complex IoT systems accessible through applications. IoT security orchestration implies that IoT assets are known to the SOAR platform. Consequently, there is an information flow from IoT assets to the SOAR platform. In the opposite direction, incident response measures target IoT assets.

Middleware. The SOAR4IoT framework integrates middleware. Besides digital twins, other middleware concepts (e.g., IoT platforms) exist. The middleware is located between IoT assets and the SOAR platform. We argue that middleware is beneficial for abstracting IoT assets. Also, IoT asset data is aggregated. Digital twins, in particular, provide semantic features (e.g., modeling components), a dedicated interface, and different perspectives (e.g., data views)

ARES 2022, August 23–26, 2022, Vienna, Austria

Empl et al.

for orchestration and response. In our case, digital twins offer a comprehensive summary of the asset’s (security) state and enable the validation of security measures.

SOAR platform. The SOAR4IoT framework contains a SOAR platform at its core. Most importantly, the SOAR platform emphasizes playbooks and their automation but includes other typical features such as ticketing, user interface, and user management. Data flows from the middleware and connected security tools to the SOAR platform for security orchestration. Then, appropriate incident response measures are disseminated.

Security tools. The SOAR4IoT framework includes security tools. Security tools (e.g., SIEM – Security Information and Event Management systems or IDS – Intrusion Detection Systems) are queried or actively provide security-relevant information. Various Cyber Threat Intelligence sources (e.g., CTI feeds) can also provide input to the SOAR platform and serve as a trigger to response actions. However, incident response actions also address security tools (e.g., updating SIEM rules or disseminating CTI).

4 PROOF OF CONCEPT

We implement the SOAR4IoT framework to validate its feasibility. Defining two use cases, we represent security measures in two playbooks (Section 4.1 and 4.2). More specifically, our experimental setup includes the SOAR platform, replication-based digital twin middleware, and IoT assets (Section 4.3). Further, we demonstrate security orchestration, automation, and incident response and show experimental results (Section 4.4). At last, we conclude our proof of concept by discussing the impact and limitations (Section 4.5).

4.1 Mirai Botnet – Use Case 1

The Mirai malware is scanning IoT devices for vulnerabilities. The attacker’s goal is to use the IoT devices for malicious purposes. Consequently, IoT assets need to be secured at the device level. This scenario represents our first use case. The following SOAR playbook describes courses of action to address Mirai-like situations that require firmware updates.

Playbook 1 Mirai Botnet (proactive)

```

1: procedure MIRAI
2:    $a \leftarrow$  IoT devices
3:   for all  $d \in a$  do
4:      $e \leftarrow$  CTI for  $d$ 
5:     if  $isVulnerable(e, d)$  and  $d.checkFirmware()$  then
6:        $d.updateFirmware()$ 
7:   if  $checkAuthentication(e, a)$  then
8:      $changeAuthentication(a)$ 
9:      $a.permitJoin(true, 30s)$ 

```

Organizational security operations to cope with Mirai or similar malware include threat intelligence. Organizations monitor their IoT devices and pay attention to vulnerabilities. Either manually or automated, organizations analyze CTI reports. CTI describes severe vulnerabilities and triggers security operations. Such security operations include checking affected IoT device status and whether

a new firmware update is available. This procedure is necessary to keep IoT devices secure and ensure continuous operation. Otherwise, IoT devices can easily contribute to malicious activities, such as distributed denial of service (DDoS) attacks.

4.2 Sybil Attack – Use Case 2

A Sybil attack in IoT describes the fake creation of identities (i.e., IoT assets) in IoT networks [26]. Thereby, attackers attempt to forward data selectively, drop data packets or manipulate data. Consequently, IoT assets need to be secured at the network level. This scenario represents our second use case. The following SOAR playbook describes courses of action to address Sybil attack situations that require device removal.

Playbook 2 Sybil Attack (reactive)

```

1: procedure SYBIL
2:    $e \leftarrow$  SIEM event
3:    $a \leftarrow$  IoT network
4:   for all  $n \in a$  do
5:     if  $isSybilNode(e, n)$  then
6:        $a.removeDevice(n)$ 
7:    $a.permitJoin(false)$ 

```

Organizational security operations to cope with a Sybil attack center on adequate monitoring of additional edge nodes or other IoT network components. Digital twins include detailed information about trusted IoT assets. Thus, they can be leveraged once a trigger (e.g., a SIEM event containing the loss of several data packets) from a security tool is received. Assessing the IoT network components, organizations can identify additional fake nodes or even missing ones and start response measures. This procedure is crucial to avoid malfunctioning IoT applications.

Multiple attacks on IoT assets demand SOAR capabilities. We opted for the two exemplary use cases based on the Mirai botnet and Sybil attack to document our SOAR4IoT framework implementation. Next, we describe our technological setup, including hardware and software.

4.3 Experimental Setup

Our experimental setup implements the SOAR4IoT framework. The source code is available in Gitlab². Figure 3 describes our prototypical implementation and documents technology and data flows. This overview is further specified by categorizing and listing the underlying hardware (see Table 3).

IoT assets. We deploy two Xiaomi Aqara temperature sensors and two IKEA Tradfri LED bulb actuators in our lab environment. The sensors measure temperature and humidity. The actuators control brightness, color temperature, and state of connected LEDs. For communication purposes, sensors and actuators use the Zigbee protocol. Additionally, we deploy a Raspberry Pi 3B+ edge node. Zigbee communication between IoT assets and the edge node is controlled with a CC2531 Zigbee USB-Stick. This Zigbee controller is physically plugged into the edge node, but communication is

²<https://git.ur.de/soar4iot>

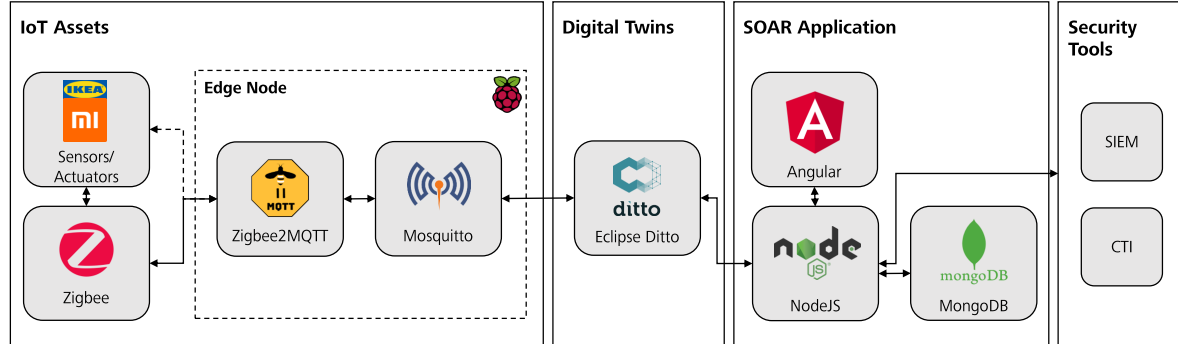


Figure 3: Experimental setting

Table 3: Hardware list

Device	Category	Characteristics
Xiaomi Aqara Temperature	Sensor	WSDCGQ01LM, Zigbee protocol
IKEA Tradfri LED Bulb E14	Actuator	LED1733G7, Zigbee protocol
CC2531 Zigbee USB flash drive	Controller	USB interface, Zigbee protocol
Raspberry Pi 3B+	Edge Node	Raspbian GNU/Linux 11, 1GB RAM, RJ-45 Ethernet
Virtual Machine	Server	Ubuntu 20.04.3 LTS, 16GB RAM, 8 cores, 80GB disc

wireless. At the edge node, the Zigbee data is transformed into MQTT data using the Zigbee2MQTT³ bridge. Zigbee2MQTT acts as a client sending data from sensors and actuators to the MQTT broker. In our setup, the open-source MQTT broker Mosquitto⁴ is installed on the edge node. As MQTT data is structured in topics, Zigbee2MQTT publishes/subscribes to an IoT asset-specific topic (e.g., SOAR4IoT/Lab_Actuator_Bulb1). In the same way, Mosquitto uses MQTT topics for upstream data. Similar IoT assets and edge nodes to our experimental setup might be used as part of an industrial oven or assembly line.

Digital twins. We implement digital twins representing the middleware of our SOAR4IoT framework. For each IoT asset there is one digital twin. Using the open-source digital twin software Eclipse Ditto⁵ allows us to integrate and replicate heterogeneous IoT assets. Eclipse Ditto enables message-oriented communication with IoT assets through their digital twin. Besides, it supports the definition of policies (i.e., access control) and the integration of specific brokers for several IoT protocols (e.g., MQTT, AMQP, or CoAP). In our experimental setup, Eclipse Ditto runs on a virtual machine

³<https://www.zigbee2mqtt.io>

⁴<https://mosquitto.org>

⁵<https://www.eclipse.org/ditto>

(Ubuntu, 16GB RAM, 8 kernels, and 80GB storage) and connects to Mosquitto.

We design and configure our Eclipse Ditto-based digital twins (see Figure 4). First, we define the primary policy. This policy grants an admin user read and write access to the digital twins and restricts a demo user to read access only. We then create five IoT assets, including the edge node. Each IoT asset is structured using JSON data serialization that defines a primary data schema for its digital twin. We further define messages in Eclipse Ditto. These messages allow users to interact directly with the digital twin of an IoT asset. Digital twins process all messages received from users separately and behave according to the message-defined function. However, not all messages are equally feasible for all IoT assets. While sensors and actuators implement firmware and state/effect messages, the edge node (network administrator) can remove or permit devices to join the network. For instance, if a new IoT asset is invited to onboard the network, the edge node temporarily allows new devices to join for 20 seconds by messaging `permitJoin(true,20)`. Last, we connect Eclipse Ditto to the Mosquitto MQTT broker to establish bidirectional communication between the digital twins and the IoT assets. On the one side, data received from the MQTT broker fills the pre-defined data schemata of the digital twins, and on the other side, digital twins can send commands to the IoT assets.

We opted for Eclipse Ditto because event-based middleware is most qualified for real-time data processing [6] and SOAR use cases. Eclipse Ditto implements the publish/subscribe approach with topics and events (see Figure 1). Nevertheless, there are several ways of implementing digital twins (e.g., physics-based modeling vs. data-driven techniques). Eclipse Ditto uses a data-driven technique with messages to represent IoT asset functions. This type of middleware fits SOAR best, as SOAR does not require simulation capabilities and other aspects of physics-based digital twins. Additionally, Eclipse Ditto is established and used by industrial companies (e.g., Bosch or Aloxly).

SOAR application. The SOAR platform application is deployed on the same virtual machine that runs Eclipse Ditto. We implemented the frontend of the SOAR platform using the Angular⁶

⁶<https://angular.io>

ARES 2022, August 23–26, 2022, Vienna, Austria

Empl et al.

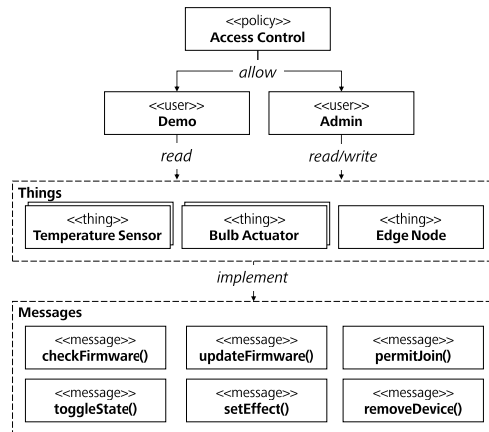


Figure 4: Digital twin setting in Eclipse Ditto

web application framework and Typescript⁷. The backend of our SOAR application is based on NodeJS⁸ storing data in a MongoDB⁹ database. Developing the SOAR application, we find microservice architecture to fit the purposes of SOAR best. Our SOAR4IoT implementation integrates four main microservices: core app (central microservice), Eclipse Ditto app, CTI app, and a SIEM app. The SIEM app generates pseudo-events used to trigger the execution of playbooks. The CTI app queries vulnerabilities, and the Eclipse Ditto app integrates IoT assets. For ease of deployment, we use Docker Compose and Docker Images. A detailed description of the SOAR platform features is described in Section 4.4.

Security tools. At last, our experimental setup includes the use of security tools. We pursue a twofold approach. First, we integrate existing CTI sources for security-relevant information. Thus, information about vulnerabilities in applications, hardware, or operating systems can be queried from the US National Vulnerability Database (NVD) and is structured by its Common Vulnerabilities and Exposures (CVE) enumeration. CVE descriptions are particularly relevant as attackers widely use available exploits for known vulnerabilities. Also, firmware update information can be queried for our actuators. Second, we directly include a security event feature. This feature is based on predefined security events representing SIEM alarms or incident notifications. Contrary to our experimental setting, organizations will integrate their existing SIEM systems or ticketing systems instead.

4.4 Results

Our research yields results concerning the demonstration of two IoT security use cases. Implementing our digital twins and IoT-centric SOAR application enables security workflows based on user interface (UI) and playbook execution.

⁷<https://www.typescriptlang.org>

⁸<https://nodejs.org>

⁹<https://www.mongodb.com>

We created three playbooks, of which two are addressing the Sybil attack and one the Mirai botnet use case. Therefore, our UI¹⁰ includes an intuitive playbook editor for configuration. In general, the UI of our SOAR application follows a minimalistic approach and provides a single point of contact. Figure 5 documents three main views: (a) security event list, (b) IoT assets (digital twins), and (c) playbooks. Our digital twin and security-focused UI goes beyond the generic Mosquito UI and the Ansible Semaphore UI¹¹. We reason that designing and implementing a customized SOAR application along SOAR requirements is feasible with open-source technologies.

We define a generic SOAR4IoT workflow to showcase playbook execution. The workflow involves IoT assets (digital twins), apps, actions, playbooks, and events. Apps (i.e., individual microservices) implement specific actions (e.g., API calls) relevant for security operations. These actions are then structured and instantiated within playbooks. At last, given a specific security event (received by app or created via the UI), playbook execution is triggered. Playbook execution is dependent on event parameters and matching logic. As events are linked to IoT assets, matched playbooks must refer to the same IoT assets. During playbook execution the SOAR core service checks an app's availability, documents action status and starts subsequent actions. The playbook status indicates success, timeout or failure.

The *Mirai* *playbook* is used for vulnerable IoT assets (e.g., missing updates or default passwords). Its actions include fetching CTI data, updating IoT assets OTA, and requesting analysts to check the IoT assets' authentication manually. Our experimental setup includes no vulnerable IoT assets, so we define a repetitive update event. This event triggers playbook execution regularly. We successfully achieved firmware updates for the IKEA Tradfri LED bulb using digital twin messaging functions and Zigbee2MQTT. Changing authentication and validating playbook execution (e.g., comparing firmware versions) are subsequent manual tasks.

The *Sybil* *playbooks* address rogue devices. The actions include identifying and removing Sybil nodes from the network. This is followed by preventing new devices to join the network. Leveraging our SIEM app, we create events indicating a possible Sybil attack. In SOAR4IoT, the security analyst can then execute a playbook to analyze IoT assets not represented by a digital twin. If so, new removal events are created and listed with the asset's network address (see Figure 5a). A security analyst can also check manually if the network address is linked to a known IoT asset (see Figure 5b). The analyst is assisted in resolving the removal event by executing the corresponding playbook (see Figure 5c). Observing the status of playbook execution, the Sybil node is successfully removed. Validation might include comparing connected IoT assets at the edge node before and after playbook execution. In general, playbook selection depends on analyst's assessment of whether a playbook's actions meet the desired objective.

Lessons Learned. We learned that a logical separation of security orchestration and incident response using microservices benefits the SOAR application. We consider security orchestration a data collection task (e.g., querying device status or available CTI) and

¹⁰<https://soar4iot.ur.de>

¹¹<https://github.com/ansible-semaphore/semaphore>

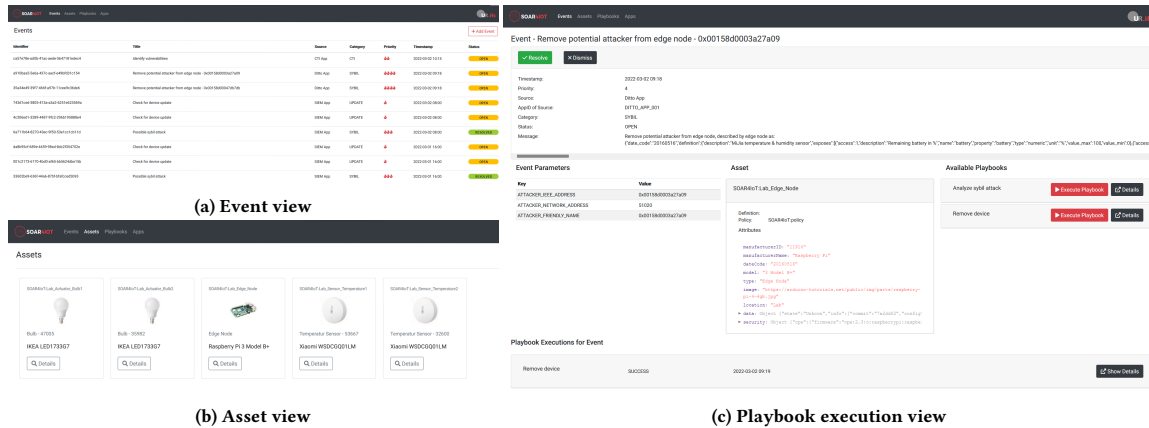


Figure 5: SOAR4IoT UI

incident response a modification task. Digital twins prove relevant as they provide an additional layer with unified access and control to make the IoT manageable for security purposes. We experienced excellent feedback from the Eclipse Ditto community during development. It leads us to conclude that, in practice, digital twins go beyond the functional scope of IoT platforms, and digital twin research is relatively narrow. Overall, SOAR application development is a challenging task, but complexity can be reduced (e.g., via microservices, virtualization, and deployment tools).

4.5 Discussion

We discuss both the scientific and practical impact of our SOAR4IoT framework before mentioning limitations.

Scientific impact. Only a few academic works have addressed security orchestration and SOAR platforms. Our work is an attempt towards leveling the playing field with the large number of commercial SOAR platforms. This attempt includes a list of SOAR platform features. Eventually, documented by our SOAR application, open-source technologies can be used. We contrast user reluctance with the potential use cases for security and open-source frameworks like Eclipse Ditto digital twins. We direct attention to digital twins for security operations beyond current simulations.

Practical impact. To cope with the current IoT trend, organizations must manage IoT assets and extend existing SOAR platforms. Our work can be seen as an innovative approach using open-source technologies. Pointing at the benefits of small-scale, customized SOAR platforms, we contrast commercial SOAR platforms. Our publicly available source code can serve for future extensions.

Limitations. There are several aspects that our work does not address. We attempted to select appropriate technologies and justify our decisions, but there are no best practices for digital twins in cybersecurity. CPS Twinning¹² is an alternative digital twin framework worth further investigation. Additionally, we excluded

¹²<https://github.com/sbaresearch/cps-twinning>

security for IoT cloud applications (e.g., predictive maintenance) typically used with IoT assets. Our SOAR application does not consider communication features (e.g., messaging or task delegation) found in commercial SOAR platforms. Access control, available for digital twins, is missing at SOAR application level but is required in production environments. Due to the small quantity of IoT assets, we can not assess the scalability of our SOAR application. Since many IoT devices will never experience updates, organizations should pay attention when buying them. Also, we did not exploit the full range of possibilities as our SOAR application integrates only a few security tools.

5 CONCLUSION AND FUTURE WORK

The question “How to use Security Orchestration, Automation and Response for the Internet of Things?” was the starting point of our work. While investigating the SOAR concept and SOAR platforms, we derived a detailed understanding of SOAR and its requirements. Defined by its orchestration, automation, and incident response capabilities, SOAR is mainly centered on playbooks and security tool integration for security operations. Extending the security operations to the IoT is a necessary step, as IoT attacks and IoT objectives show. Among different options to secure the IoT, digital twins provide a feasible, lightweight solution abstracting heterogeneous assets. Thus, our SOAR4IoT framework integrates a digital twin-based middleware. More precisely, we establish a prototypical implementation using Eclipse Ditto and a microservice SOAR application. Implications of our conceptual design and SOAR4IoT implementation include the following:

- Digital twins provide abstraction and a unified interface for the plethora of IoT assets. The security community should further compare different digital twin frameworks’ abilities (e.g., advanced behavior or process modeling). To the best of our knowledge, our Eclipse Ditto implementation is the first, with security use cases built on top. It can serve as a

stepping stone for sophisticated intrusion detection, threat notifications, and life cycle analyses.

- SOAR is about playbooks. Thus, research should focus on the great potential of playbooks. We expect benefits of identifying additional use cases (e.g., execution of playbooks against a group of IoT assets) and formalizing playbook logic. Future work should assist security analysts from initial (automated) playbook creation based on manufacturers' course of action recommendations to playbook \times event matching and (prioritized) execution. Therefore, playbooks must consider organizational incident response processes and their underlying principles.

From a security management perspective, SOAR4IoT has two great strengths. First, it is crucial to see the full picture and properly manage organizational assets. And second, security management must plan security operations strategically to maintain the security posture. Digital twins and SOAR playbooks foster both aspects. However, this requires initial resources to implement the SOAR4IoT framework and strategic decisions whether to use playbooks to their full extent. We believe it is worth the effort due to new avenues and security possibilities.

ACKNOWLEDGMENTS

This research was supported by the German Federal Ministry of Education and Research (BMBF) as part of the DEVISE project. This research was supported by the German Federal Ministry for Economic Affairs and Climate Action (BMWK) as part of the Secure Industrial Semantic Sensor Cloud (SISSec) project.

REFERENCES

- [1] Ala I. Al-Fuqaha, Mohsen Guizani, Mehdi Mohammadi, Mohammed Aledhari, and Moussa Ayyash. 2015. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Communications Surveys & Tutorials* 17, 4 (2015), 2347–2376. <https://doi.org/10.1109/COMST.2015.2444095>
- [2] Jan Bauwens, Peter Ruckebusch, Spiros Giannoulis, Ingrid Moerman, and Eli De Poorter. 2020. Over-the-Air Software Updates in the Internet of Things: An Overview of Key Principles. *IEEE Communications Magazine* 58, 2 (2020), 35–41. <https://doi.org/10.1109/MCOM.001.1900125>
- [3] Stefan Boschert, Christoph Heinrich, and R. Rosen. 2018. Next Generation Digital Twin. In *Proceedings of the 12th. International Symposium on Tools and Methods of Competitive Engineering (TMCE'18)* (Las Palmas de Gran Canaria, Spain), I. Horvath, J.P. Suarez Riviero, and P.M. Hernandez Castellano (Eds.), 209–218.
- [4] Ismail Butun, Patrik Österberg, and Houbing Song. 2020. Security of the Internet of Things: Vulnerabilities, Attacks, and Countermeasures. *IEEE Communications Surveys & Tutorials* 22, 1 (2020), 616–644. <https://doi.org/10.1109/COMST.2019.2953364>
- [5] Seoyun Choi and Jong-Hyook Lee. 2020. Blockchain-based distributed firmware update architecture for IoT devices. *IEEE Access* 8 (2020), 37518–37525. <https://doi.org/10.1109/ACCESS.2020.2975920>
- [6] Mauro A. A. da Cruz, Joel José Puga Coelho Rodrigues, Jalal Al-Muhtadi, Valery Korotaev, and Victor Hugo C. de Albuquerque. 2018. A Reference Model for Internet of Things Middleware. *IEEE Internet of Things Journal* 5, 2 (2018), 871–883. <https://doi.org/10.1109/JIOT.2018.2796561>
- [7] Violeta Damjanovic-Behrendt. 2018. A digital twin architecture for security, privacy and safety. *ERCIM News* 115 Special Issue "Digital Twins (2018)".
- [8] Marietheres Dietz, Manfred Vielberth, and Günther Pernul. 2020. Integrating digital twin security simulations in the security operations center. In *Proceedings of the 15th International Conference on Availability, Reliability and Security (ARES'20)* (Virtual Event), Melanie Volkamer and Christian Wressnegger (Eds.), 18:1–18:9. <https://doi.org/10.1145/3407023.3407039>
- [9] Matthias Eckhart and Andreas Ekelhart. 2019. Digital twins for cyber-physical systems security: State of the art and outlook. *Security and quality in cyber-physical systems engineering* (2019), 383–412.
- [10] Matthias Eckhart, Andreas Ekelhart, and Roland Eisl. 2021. Digital Twins for Cyber-Physical Threat Detection and Response. *ERCIM News* 127 (2021).
- [11] Forrester Consulting. 2020. *The 2020 State Of Security Operations*. Technical Report E-46260. Forrester Research (commissioned by Palo Alto Networks), Cambridge, England.
- [12] Janis Grabis, Janis Stima, and Jelena Zdravkovic. 2021. A Capability Based Method for Development of Resilient Digital Services. In *Enterprise Information Systems*, Joaquim Filipe, Michal Šmialek, Alexander Brodsky, and Slimane Hammoudi (Eds.), Vol. 417, 498–516. https://doi.org/10.1007/978-3-030-75418-1_23
- [13] Chadni Islam, Muhammad Ali Babar, and Surya Nepal. 2019. A Multi-Vocal Review of Security Orchestration. *Comput. Surveys* 52, 2, Article 37 (2019), 45 pages. <https://doi.org/10.1145/3305268>
- [14] Chadni Islam, Muhammad Ali Babar, and Surya Nepal. 2020. Architecture-Centric Support for Integrating Security Tools in a Security Orchestration Platform. In *Proceedings of the 14th. European Conference on Software Architecture (ECSA'20)* (L'Aquila, Italy), A. Jansen, I. Malavolta, H. Muccini, I. Ozkaya, and O. Zimmermann (Eds.), Springer, Cham, Germany, 165–181. https://doi.org/10.1007/978-3-030-58923-3_11
- [15] Bernd Jäger. 2015. Security Orchestrator: Introducing a Security Orchestrator in the Context of the ETSI NFV Reference Architecture. In *Proceedings of the 14th. IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom'15)* (Helsinki, Finland), IEEE, New York, NY, USA, 1255–1260. <https://doi.org/10.1109/Trustcom.2015.514>
- [16] Xingwei Liang and Yoohwan Kim. 2021. A Survey on Security Attacks and Solutions in the IoT Network. In *Proceedings of the 11th. IEEE Annual Computing and Communication Workshop and Conference (CCWC'21)* (Virtual Event), IEEE, New York, NY, USA, 853–859. <https://doi.org/10.1109/CCWC51732.2021.9376174>
- [17] Parushi Malhotra, Yashwant Singh, Pooja Anand, Deep Kumar Bangotra, Pradeep Kumar Singh, and Wei-Chiang Hong. 2021. Internet of Things: Evolution, Concerns and Security Challenges. *Sensors* 21, 5 (2021), 1809. <https://doi.org/10.3390/s21051809>
- [18] Brendan Moran, Hannes Tschofenig, David Brown, and Milosch Meriac. 2021. *A Firmware Update Architecture for Internet of Things*. Technical Report. RFC 9019. Internet Engineering Task Force (IETF).
- [19] Claudio Neiva, Craig Lawson, Toby Bussa, and Gorka Sadowski. 2020. *2020 Market Guide for Security Orchestration, Automation and Response Solutions*. Technical Report. Gartner.
- [20] Netscout. 2020. *Netscout Threat Intelligence Report (Issue 6)*. Technical Report. Netscout.
- [21] Savannah Norem, Ashley E Rice, Samantha Erwin, Robert A Bridges, Sean Oesch, and Brian Weber. 2021. A Mathematical Framework for Evaluation of SOAR Tools with Limited Survey Data. <https://doi.org/10.48550/arXiv.2112.00100>
- [22] Megan Nyre-Yu. 2021. Identifying Expertise Gaps in Cyber Incident Response: Cyber Defender Needs vs. Technological Development. In *Proceedings of the 54th. Hawaii International Conference on System Sciences (HICSS'21)* (Wailea, Hawaii), 1978–1987.
- [23] OASIS. 2020. *Open Command and Control (OpenC2) Language Specification Version 1.0 - Committee Specification 02*. OASIS. <https://docs.oasis-open.org/openc2/oc2ls/v1.0/cs02/oc2ls-v1.0-cs02.html> Last accessed 2021-11-20.
- [24] OASIS. 2021. *CACAO Security Playbooks Version 1.0 - Committee Specification 01*. OASIS. <https://docs.oasis-open.org/cacao/security-playbooks/v1.0/security-playbooks-v1.0.html> Last accessed 2021-11-20.
- [25] Palo Alto Networks. 2020. *Measuring the ROI of an Incident Response Platform*. Technical Report UC-031220. Palo Alto Networks, Santa Clara, CA, USA.
- [26] Anjana Rajan, J. Jithish, and Sriram Sankaran. 2017. Sybil attack in IOT: Modelling and defenses. In *Proceedings of the 6th. International Conference on Advances in Computing, Communications and Informatics, ICACCI'17* (Manipal, India), IEEE, New York, NY, USA, 2323–2327. <https://doi.org/10.1109/ICACCI.2017.8126193>
- [27] Mohammad Abdur Razzaque, Marija Milojevic-Jevric, Andrei Palade, and Siobhán Clarke. 2016. Middleware for Internet of Things: A Survey. *IEEE Internet of Things Journal* 3, 1 (2016), 70–95. <https://doi.org/10.1109/JIOT.2015.2498900>
- [28] Daniel Schlette, Marco Caselli, and Günther Pernul. 2021. A Comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective. *IEEE Communications Surveys & Tutorials* 23, 4 (2021), 2525–2556. <https://doi.org/10.1109/COMST.2021.3117338>
- [29] Daniel Schlette, Florian Menges, Thomas Baumer, and Günther Pernul. 2020. Security enumerations for cyber-physical systems. In *IHIP Annual Conference on Data and Applications Security and Privacy (DBSec'20)* (Virtual Event), Springer, Cham, Germany, 64–76.
- [30] Tara Seils. 2021. IoT Attacks Skyrocket, Doubling in 6 Months. <https://threatpost.com/iot-attacks-doubling/169224/>. Last accessed 2021-02-21.
- [31] Manfred Vielberth, Fabian Bohm, Ines Fichtinger, and Günther Pernul. 2020. Security Operations Center: A Systematic Study and Open Challenges. *IEEE Access* 8 (2020), 227756–227779. <https://doi.org/10.1109/ACCESS.2020.3045514>
- [32] Jingbin Zhang, Meng Ma, Ping Wang, and Xiao-dong Sun. 2021. Middleware for the Internet of Things: A survey on requirements, enabling technologies, and solutions. *Journal of Systems Architecture* 117 (2021), 102098. <https://doi.org/10.1016/j.sysarc.2021.102098>

5 Generating ICS Vulnerability Playbooks with Open Standards

Publication information

Current status:	Under review								
Conference:	18th International Conference on Availability, Reliability and Security, Benevento, Italy, August 29 - September 01, 2023								
Date of submission:	15 March 2023								
Full citation:	EMPL, P., SCHLETTE, D., STÖGER, L., & PERNUL, G. (2023). Generating ICS Vulnerability Playbooks with Open Standards. Submitted to <i>The 18th International Conference on Availability, Reliability and Security (ARES 2023)</i> .								
Authors' contributions:	<table><tr><td>Philip Empl</td><td>40%</td></tr><tr><td>Daniel Schlette</td><td>40%</td></tr><tr><td>Lukas Stöger</td><td>10%</td></tr><tr><td>Günther Pernul</td><td>10%</td></tr></table>	Philip Empl	40%	Daniel Schlette	40%	Lukas Stöger	10%	Günther Pernul	10%
Philip Empl	40%								
Daniel Schlette	40%								
Lukas Stöger	10%								
Günther Pernul	10%								

Conference description: The International Conference on Availability, Reliability and Security (ARES) brings together researchers and practitioners in the field of IT security & privacy. Since 2005, ARES serves as an important platform to exchange, discuss and transfer knowledge and is hosted every year in another European city.

Generating ICS Vulnerability Playbooks with Open Standards

Anonymous Author(s)

ABSTRACT

Organizations face attacks on industrial control systems (ICS) as vulnerabilities are pervasive. However, patching vulnerable systems by simply updating to the newest version is often not an option and shifts focus to workarounds. Beyond pure patching, workarounds specify other remediation measures (e.g., firewall or VPN configuration) that must be taken due to system availability requirements, complexity, or heterogeneous devices. In this paper, we introduce vulnerability playbooks based on open standards. Pushing the envelope of cybersecurity playbooks and their step-by-step instructions to ICS vulnerability management offers organizations a more transparent, repeatable process and faster, possibly automated actions. We have designed a process model to collect and transform security advisories in *Common Security Advisory Framework* (CSAF) format and generate *Collaborative Automated Course of Action Operations* (CACAO) playbooks based on listed remediation advice. With a proof of concept, we demonstrate that structured CSAF documents can be seamlessly transformed into CACAO playbooks. For our industrial use case, we must also use unstructured security advice highlighting quality differences (compared to CSAF). Our generated 79 standard-conformant CACAO playbooks with 485 identified actions hint at imbalanced advice towards patching. Preferably, vendors should include detailed technical remediation advice and go beyond patching recommendations in their security advisories.

CCS CONCEPTS

• Security and privacy → Vulnerability management; • Information systems → Information systems applications.

KEYWORDS

Security advisory, playbook, vulnerability, ICS, CSAF, CACAO

ACM Reference Format:

Anonymous Author(s). 2023. Generating ICS Vulnerability Playbooks with Open Standards. In *The 18th International Conference on Availability, Reliability and Security (ARES 2023), August 29– September 01, 2023, Benevento, Italy*. ACM, New York, NY, USA, 11 pages. <https://doi.org/10.1145/XXXXX.XXXXX>

1 INTRODUCTION

Cybersecurity playbooks are about knowing what to do when insecurity becomes apparent. The heavily promoted notion of playbooks captures the description of organizational processes, specified workflows, and individual actions. Security Orchestration, Automation and Response (SOAR) tools rely on playbooks [14], and the US

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

ARES 2023, August 29– September 01, 2023, Benevento, Italy

© 2023 Copyright held by the owner/author(s).

ACM ISBN XXXXX

<https://doi.org/10.1145/XXXXX.XXXXX>

government, special interest groups, and researchers are eager to develop playbooks [13, 22, 28]. With industry support, the *Collaborative Automated Course of Action Operations* (CACAO) playbook format aims to standardize playbooks upholding the principle of open standards [20].

Existing playbooks often address incident types (e.g., phishing or malware) and research is focused on incident response [28]. However, using playbooks to handle specific vulnerabilities is another promising field that vulnerability management tools have only partially explored. ICS vulnerability playbooks can fill the gap and provide additional remediation advice to organizations. We make a first approach to answer the question: Is it possible to generate ICS vulnerability playbooks?

Our work focuses on vulnerability playbooks for ICS and the industrial IoT. These systems are affected by numerous vulnerabilities and countless attacks. For instance, according to the National Vulnerability Database (NVD), 72 vulnerabilities for SIMATIC S7 were discovered in the last ten years and caused the vendor to issue patches and security advisories. Moreover, ICS high-availability requirements, complexity, and many heterogeneous devices complicate (manual) vulnerability management and demand measures beyond updating [29]. Thus, ICS vendors typically offer security advisories detailing workarounds for remediation when system availability is a must and patching is not an option. In addition, the US Cybersecurity & Infrastructure Security Agency (CISA) maintains a collection of ICS advisories [5].

Looking at security advisories, we see different vendors use different data formats. One such format is the *Common Security Advisory Framework* (CSAF), an open standard foreseen to exchange machine-readable information [21]. It includes a dedicated section on remediation options which builds the basis for our streamlined, automated vulnerability playbook generation. Organizations can benefit from ICS vulnerability playbooks by reducing the manual handling of workarounds in multiple ways. Most notably, organizations can limit error-prone information extraction and structuring. Automating the process further increases process transparency and data provenance. These improvements are typically associated with playbooks, which leads us to create vulnerability playbooks based on security advisories.

In this work, we design and implement a process model on top of open standards for security advisories (i.e., CSAF) and playbooks (i.e., CACAO) to generate ICS vulnerability playbooks. We aim at demonstrating the practical benefits of structured security advisories making both security advisory publishers and consumers aware of this. In particular, we leverage public advisory sources and preprocess their data. Thereby, we model devices representing Siemens and Cisco assets. In our proof of concept implementation, we query security advisories from two leading ICS vendors and CISA relevant to our use case. In total, we generate 79 vulnerability playbooks and identify 485 workflow actions. Matching terms, which can be customized, are used to classify playbook steps containing the workflow actions. Our main contributions are:

- A process model for generating ICS vulnerability playbooks. The process model covers four phases: 1) querying vulnerability information, 2) sourcing security advisories, 3) converting data in CSAF, and 4) leveraging matching terms to create CACAO playbooks with workflow actions.
- An open-source application¹ to generate vulnerability playbook with open standards and industry use case.
- Recommendations for improvement and use of security advisory and playbook standards.

The paper is structured as follows. In Section 2, we present a motivating ICS vulnerability and the associated security advisory. Additional background on open standards for vulnerabilities, incident response playbooks, and related work is part of Section 3. Section 4 details our process model automating the creation of vulnerability playbooks for ICS. Then, we evaluate our approach with a use case and open-source tool implementation in Section 5. In Section 6, we outline recommendations for better vulnerability handling. In Section 7, we conclude with future research directions.

2 MOTIVATION

We illustrate the representation of security advisories with a highly critical (CVSS base score of 10) ICS vulnerability affecting Siemens SIMATIC CP devices, communication processors used in digital factories [18]. Identified by CVE-2022-34819, the vulnerability centers on improper input validation and the resulting heap-based buffer overflow. As a consequence, attackers could execute malicious code and cause production to halt. We use this vulnerability to emphasize aspects of ICS security advisories and their representation in CSAF format. Figure 1 shows the abbreviated CSAF document. Upfront metadata informs about the CSAF format and the security advisory publisher, typically the vendor of the affected product(s). A string-based title is used to refer to the security advisory. However, product users are mostly interested in security advisories to extract relevant information on vulnerability remediation. Therefore, crucial remediation advice in CSAF is listed inside a remediations array. Besides vendor fixes instructing to update to the newest version (omitted for brevity in Figure 1), workarounds detail alternative remediation steps. These workarounds help to harden SIMATIC CP devices until patches are installed. In the example CSAF, these include blocking access to a specific port by using an external firewall and disabling a VPN feature.

Security advisories and (if available) their CSAF documents do not always contain detailed and executable information. The CSAF example in Figure 1 represents a best-case scenario. Subscribers are faced with security advisories in various data formats, which are often not machine-readable. Considering heterogeneous devices, multiple vulnerabilities, and security advisory sources, automated vulnerability playbooks generation is evident. In contrast to manual advisory processing, process consistency can be improved. For instance, the manual vulnerability handling is error-prone or takes even more time. We would like to emphasize that the best-case scenario is not always given. When automatically creating CACAO playbooks from security advisories, we must also deal with unstructured remediation advice until the CSAF standard is established.

¹<https://www.github.com/ad2play/ad2play>

```
{
  "document": {
    "category": "csaf_security_advisory",
    "csaf_version": "2.0",
    "publisher": {
      "category": "vendor",
      "name": "Siemens ProductCERT"
    }
  },
  "title": "SSA-517377: Multiple Vulnerabilities in the
    SRCS VPN Feature in SIMATIC CP Devices"
},
"vulnerabilities": [
  {
    "cve": "CVE-2022-34819",
    "remediations": [
      {
        "category": "workaround",
        "details": "Block access to port 5243/udp e.g.
          with an external firewall if possible"
      },
      {
        "category": "workaround",
        "details": "Disable the SINEMA Remote Connect
          Server (SRCS) VPN feature"
      }
    ]
  }
] ...
```

Figure 1: Excerpt of a CSAF document for CVE-2022-34819 with remediation advice that specifies two workarounds.

3 BACKGROUND AND RELATED WORK

Open standards for vulnerability management and incident response playbooks represent foundations for our work. We further discuss related work within this section.

3.1 Open Security Standards

Vulnerability management relies on a shared understanding of concepts. Open security standards provide the means to cope with low information quality by assisting with uniform representation and content structure. The following standards and data formats are widely recognized in cybersecurity and help organizations handle vulnerabilities.

CVE – Common Vulnerability Enumeration, used to identify and describe vulnerabilities.

CPE – Common Platform Enumeration, used to identify IT/OT assets.

CVSS – Common Vulnerability Scoring System, used to define and assign severity scores.

CVRF/CSAF – Common Vulnerability Reporting Framework/Common Security Advisory Framework, used to describe security advisories.

The open standards and data formats are intended to inform others about vulnerabilities, exploits, and remediation advice [27]. They answer the crucial questions: what characterizes a vulnerability? What systems are affected? How severe is the vulnerability? And what do others need to know about vulnerability remediation?

3.2 Incident Response Playbooks

Organizations need to define processes, procedures, and actions for incident response. Threat intelligence is also necessary to handle security incidents [10]. Thus, incident response representations

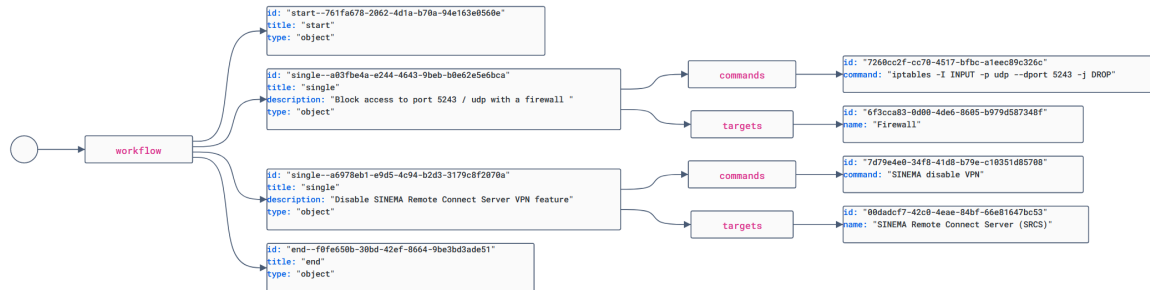


Figure 2: Schematic visualization of a CACAO vulnerability playbook that includes workflow, command, and target objects.

with playbooks bridge the gap between processes and data containing both [24]. Mainly two major use cases – the automation of incident response and the sharing of playbooks – have resulted in the development of open standards and data formats (e.g., CACAO, OpenC2, MITRE D3FEND, or RE&CT) for playbooks and individual actions [17, 19, 23].

CACAO. Collaboratively developed by the Organization for the Advancement of Structured Information Standards (OASIS) and its members, the open CACAO format targets playbooks. In contrast to more action-focused standards, CACAO playbooks can describe information on various granularity levels. As a result, the CACAO format is comprehensive and a promising candidate for the description of vulnerability playbooks. To the best of our knowledge, there are no other maintained and open playbook standards with similar characteristics. Using CACAO playbooks, organizations can follow defined workflows and have the ability to automate repetitive, error-prone tasks.

The benefits of the CACAO playbook format are best understood by looking at its structure and object types. Figure 2 shows the visualization of a vulnerability playbook for CVE-2022-34819 and the underlying attribute-value pairs in JSON. The playbook is based on real-world vendor remediation advice augmented with commands. CACAO playbooks contain workflows to structure workflow steps. Typically, start and end steps enclose single action steps outlining specific actions. On a more granular level, command and target objects describe executable information and its recipients. In the example, organizations can derive two remediation actions, systems involved (i.e., firewall, server), and commands (i.e., iptables, disable). CACAO is broad in scope, and command and target types also support manual actions for individuals. Adding conditional workflow steps helps to represent sophisticated workflows. We use CACAO as it can capture multiple CSAF-based remediation measures and hand action-based workflows to organizations. In the remainder of this paper, we refer to workflow steps as workflow actions to differentiate between CSAF and CACAO.

3.3 Related Work

Vulnerabilities and vulnerability management are of interest to researchers and organizations alike. Organizations are advised to systematically handle vulnerabilities as they can lead to severe

security incidents [30]. A steady stream of research covers general and ICS-specific vulnerabilities [11, 25]. From vulnerability discovery [15], to vulnerability assessment [2] and security advisories [9], transparent processes and standards are important. While Fenz et al. [8] introduce automated handling of security advisories, other work has taken on the challenge to provide commit-level patch advice for vulnerabilities [4]. Besides, vulnerability management is of practical interest as vendors of commercial vulnerability management tools address the need to keep track of assets and vulnerabilities.

Academic work on cybersecurity playbooks is sparse. Nevertheless, playbooks are an emerging research topic related to threat intelligence and security standards [24]. In a recent study, Stevens et al. [28] explored human playbook creation with available frameworks indicating playbook variety. As different approaches and sharing use cases exist, integration and semantics of playbooks are investigated [1, 16, 26]. Against the backdrop of security orchestration and a plethora of commercial SOAR tools [14], specific use cases (e.g., an IoT context with digital twins) have been discussed [7, 12]. It can be seen that playbook generation is crucial to leverage SOAR tools.

We go beyond related work in the following ways. Our approach is the first to combine the two areas of security advisories and playbooks. Building vulnerability playbooks offers organizations more process-oriented advice on what to do. While some vulnerability management tools incorporate the idea of playbooks, we see benefits in following a similar path with open security standards. Our focus on ICS security advisories capitalizes on the fact that other remediation measures are most important when simply patching is not an option. Playbooks can introduce transparent processes and automation toward better vulnerability management for ICS.

4 VULNERABILITY PLAYBOOK GENERATION

Driven by the problem of ICS vulnerability handling and inspired by related works, we develop a process model. From *security advisory to vulnerability playbook*, the process captures automated ICS vulnerability playbook generation with four phases shown as a BPMN diagram in Figure 3. The subsequent sections are dedicated to the illustrated process phases.

ARES 2023, August 29– September 01, 2023, Benevento, Italy

Anon.

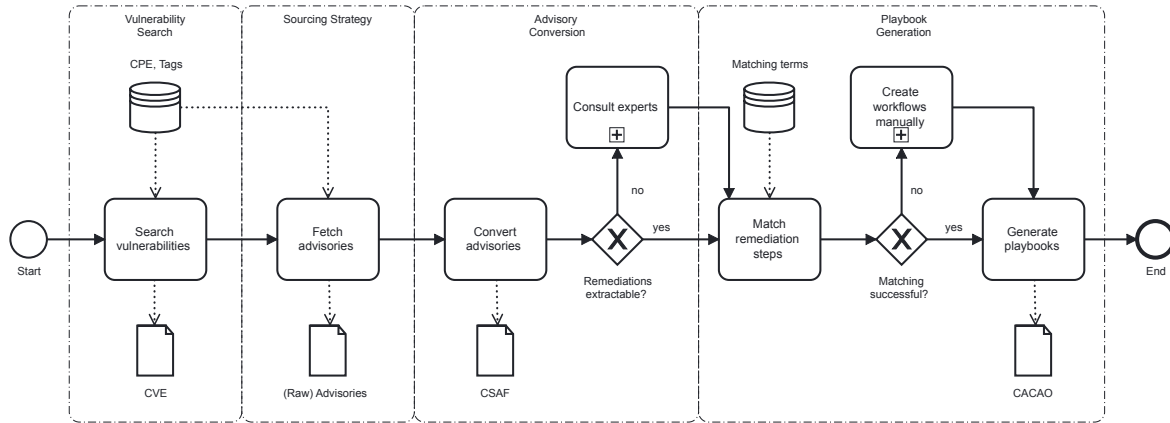


Figure 3: Process description from security advisory to vulnerability playbook.

4.1 Vulnerability Search

Vulnerability handling and the playbook generation process start with assets and the question of whether these assets are vulnerable or not. Thus, we define the activity *Search vulnerabilities* to get an overview of relevant vulnerabilities. As a prerequisite, organizations must already carefully document their assets and components (e.g. virtual representations or SBOM). Using this documentation, assets and respective identifiers (e.g., CPE-ID) are used to find vulnerabilities. However, the specific characteristics of ICS need to be considered. Most notably, ICS assets are built of multiple components forming complex systems-of-systems [6]. Each of the components can run its own software on dedicated hardware. Searching for relevant security vulnerabilities requires identifiers – for the vulnerability and the components. Vulnerabilities are given a CVE-ID. ICS components (i.e., hardware or software) have a CPE-ID or other tags. If a component is described by CPE, querying associated CVEs is straightforward. Without CPE, other information (e.g., model or version) must be used to search vulnerability databases. In our process model intended for automation, we rely on CPE or, when not available, use device-specific tags. Both approaches enable automated vulnerability searches but using tags might lead to more false positives/negatives. The first phase yields vulnerabilities regardless of available security advisories.

4.2 Sourcing Strategy

The activity *Fetch advisories* is part of the second process phase. Our sourcing strategy involves security advisory acquisition from product vendors. These product vendors often have Product Security Incident Response Teams (PSIRTs/ProductCERTs) that offer vulnerability remediation advice for their products. In addition, other institutions (e.g., national or coordination Computer Emergency Response Teams, CERTs/CSIRTs) and commercial security vendors partially aggregate security advice. In most cases, security advisories can be fetched with CPEs or tags and link to CVEs. Focusing on ICS and fetching security advisories for systems-of-systems involves multiple sources varying in format, structure, and

content. We compared security advisory publishers and data formats. Most sources provide access to PDF security advisories or embed these directly on their website. In the best-case scenario, we find dedicated formats such as CSAF or its predecessor CVRF, but they can be retrieved less often. Formats like HTML or PDF require deep traversal and scraping. Since both formats do not provide options to directly map assets to remediation advice, we need to filter whether the remediation advice actually affects devices of interest. Therefore, we recommend the device representations to skim and filter these documents automatically. As a means of communication, many of the listed sources offer RSS feeds, email notifications, or communicate the latest advice via Twitter.

The various communication channels do not solve the problem of data heterogeneity and do not always allow the exchange of remediation advice and feedback. Additionally, many sources do not provide an API to fetch security advisories for specific vulnerabilities. Automating the filtering of RSS feeds or emails to match the advisories of interest is an unnecessarily complex intermediate step. However, organizations relying on different sources and advisory formats must convert and standardize these advisories to enable automation.

4.3 Advisory Conversion

The activity *Convert advisories* targets standardization and the results are uniform security advisories. Since different vendors use different formats for representing and sharing security advisories, it is essential to convert these heterogeneous security advisories into a uniform format before generating playbooks. We rely on the open standard CSAF for the structuring and presentation of remediation steps. Thus, it is the objective of this activity to convert security advisories into CSAF documents.

There are three possible cases. First, CVRF is converted to CSAF using semantically identical fields to store remediation steps. Second, when security advisories are provided as CSAF documents, they are not converted and taken as is. However, we discard remediation steps not matching the CPE identifiers or tags. Last, also other

Table 1: Action classification and related terms based on the OpenC2 commands.

Class	Terms
Update	[["patch", "update", "upgrade"], ["version", "v", "ver"]]
Investigation	[["investigation", "investigate", "scan", "examine", "inspect", "inspection", "review", "check"]]
Locating	[["locate", "find", "detect", "discover", "uncover"], ["object", "artifact", "file", "directory", "instance"]]
Data-Operation	[["query", "create", "alter", "delete", "copy"], ["data", "entity", "directory", "file"]]
Isolation	[["contain", "containment", "isolation", "avoid"], ["file", "process", "entity", "asset"]]
Privileges	[["access", "credentials", "right"], ["allow", "restrict", "grant", "assign", "give", "permit", "reduce", "regulate", "block", "limit"]]
System	[["start", "stop", "restart", "cancel", "enable", "disable"], ["process", "application", "system", "activity", "action", "environment", "function", "feature", "port"]]
Configuration	[["set", "change", "apply", "put", "restore"], ["value", "configuration", "state", "property", "attribute"]]
Network	[["redirect", "switch", "block", "intercept"], ["traffic", "destination", "url", "ip", "port", "address", "packet", "network"]]
Observation	[["detonate", "execute", "observe", "examine", "monitor", "discover"], ["behaviour", "malware", "target", "action", "attack", "activity"]]

source-specific types of security advisory formats are converted. Here, remediation advice needs to be extracted. Dependent on the data format, steps may include HTML/PDF parsing and scraping to identify and extract nested remediation steps. For instance, in the case of CISA security advisories, we suggest to extract the mitigation section, the executive summary, and the technical details besides relevant meta data, i.e., title, date, or URL. Note that, these steps require logic to filter the remediation advice as unstructured data does not maintain a reliable mapping between remediation advice and devices of interest. In a scenario where no remediation advice is available, we include user interaction and consult experts. Possible calls to action include the search for internal playbooks targeting similar vulnerabilities. These playbooks might provide remediation steps that can fit the currently investigated vulnerability. Regardless of the scenario, this process phase results in standardized security advisories with remediation steps for vulnerability playbook generation.

4.4 Playbook Generation

After unifying security advisories, we move from security advisories to playbooks involving activities to *Match remediation steps* and *Generate playbooks*. We rely on the open standard CACAO for structuring playbook-related information and workflow actions. In CSAF, remediation steps are mostly textual descriptions that are not actionable. We aim at deriving workflow actions for playbook execution. Thereby we take remediation steps from CSAF, classify them and put appropriate predefined actions into the CACAO workflow action section.

We introduce the concept of *matching terms* deciding about the class of a specific workflow action. In advance, organizations must define and assign action templates to specific classes. This allows playbooks to be dynamically populated with respective actions matching a class. These matching terms resemble a two-dimensional search on remediation steps. One dimension describes the action and the other dimension the target. Matching both dimensions is essential to meet a stemmed matching term fully. A given string must at least match one word per dimension to reach the next one. With the approach, it is possible to define complex matching terms. We initially define the action classes based on individual actions (e.g., create, update, or delete) proposed by the

OASIS OpenC2 standard. Note that, organizations are flexible in their choice of classification, mapping and creation of specific action templates; OpenC2 is only one option to classify. The classification helps to tag the playbook accordingly. While these actions are used in our work to classify workflow steps, CACAO command objects can capture OpenC2 commands supporting agnostic automation. Table 1 showcases possible action classes and related matching terms. For full automation, organizations must create and assign action templates to specific classes and matching terms to populate the playbook dynamically. These action templates might be selected based on the matched terms and dynamically fed by variables (e.g., port = 5243 and target = firewall).

Applying matching terms to remediation steps requires disassembling these steps into sentences and understanding their intention. Natural language processing (NLP) is an accepted method to process and understand human-readable language. Breaking a remediation step into sentences and tokenizing each sentence leads to a set of words. Then, these words are brought into the basic form using stemming. Finally, the action class is identified if a stemmed matching term applies to a sentence across its dimensions. The following example demonstrates the two-dimensional mapping matching the terms "block" and "port" and resulting in the "network" action class:

Remediation step: *block access to port 5243/udp*
→ Stemming: *[block, access, to, port, 5243/udp]*
→ Matching: *[[block],[port]]*
→ Tag: *Class → network*
Suggested action: *Block port (port: 5243, target: firewall)*

As can be seen, the matching terms identify the class of a workflow action. Playbook-relevant parameters can be passed in this context. Ideally, actions should rely on predefined commands fitting match term combinations to automate the vulnerability handling completely. Towards automated execution of vulnerability playbooks, more granular action classes with more matching term combinations are necessary. Nevertheless, workflow steps are only one part of a CACAO playbook. Besides the workflow, a CACAO playbook also contains metadata and targets. The playbook generation activity places the remediation steps in the workflow section of the CACAO playbook and fills the remaining fields with metadata and additional information.

Our process model tends not to automate the whole process, from identifying a vulnerability to its remediation. We see this process model as a means to assist analysts by identifying and suggesting asset-relevant security advice. The playbook generation phase also involves two manual steps. First, if there is a matching error, e.g., no classification is possible, analysts can manually label workflow actions to continue the process. Second, the process model ends after suggesting a vulnerability playbook to the analyst. It is then up to the analysts whether they would like to execute, adjust, or delete the playbook. Of course, in a best case scenario, these steps would be automated, although it is questionable whether organizations are willed to apply remediation advice to critical assets without reviewing them.

5 EVALUATION

We show that it is feasible to seamlessly generate vulnerability playbooks from structured security advisories with a reasonable amount of effort. Additionally, we compare the quality and completeness of playbooks generated using structured and unstructured security advisories. In doing so, we implement our process model with a proof of concept satisfying a real-world industrial use case. Our use case defines two device representations to model systems-of-systems with vendor-specific components. Our application implements the *security advisory to vulnerability playbook* process aggregating remediation advice from three sources differing in data format, namely *Siemens ProductCERT* – CSAF, *Cisco Security Advisories* – CVRF, and *CISA ICS CERT* – HTML.

5.1 Industrial Use Case

Our real-world industrial use case describes an enterprise that is a market leader in plant and building technology, traffic and telecommunications systems, the process industry, and photovoltaic and wind power plants. As a manufacturing enterprise with over 2,000 employees, the ICS consists of several assets from Siemens and Cisco. The enterprise already tracks the vulnerabilities of IT assets, such as software-packages. The monitoring of vulnerabilities in the ICS is currently still under development. Tracking vulnerabilities and managing remediation advice is perceived as a mammoth task due to the heterogeneity and plethora of assets in use. The enterprise is highly interested in an automated solution gathering vulnerabilities and remediation advice for its assets.

We model virtual device representations (i.e., components, CPE identifier, and tags) detailing ICS assets in use. These device representations form a flexible construct to model complex systems-of-systems. To not reveal the assets, we have augmented them with several other products. In doing so, we created two obfuscated device representations. The first device comprises 22 Siemens components typically used in industrial environments. The second device defines 17 Cisco devices, e.g., used as gateways or controllers.

5.2 Experimental Setting

Our experimental setting consists of adequate hardware and software serving the industrial use case. We have implemented an application with a user interface to efficiently integrate analysts into the vulnerability playbook generation process.

We run all experiments on a single virtual machine with Ubuntu 22.04 LTS operating system, 8GB RAM and 80GB storage. The device representations are structured using JSON, similar to the widely used Eclipse Ditto² representation. The application is based on a front-end/back-end architecture and fully conforms to the CSAF and CACAO standards. The front-end is based on Vue.js, and the back-end on Node.js. The front-end is the entry point for the user to verify the correct processing of the security advisories. It provides several functions: CSAF and CACAO visualization, task overview and execution, matching term management, a CSAF converter, and a playbook configurator. A task³ is considered open if no workflow actions can be derived. A task is done when the workflow actions have been successfully processed, but the final human assessment and approval are pending. The back-end relies on the model-view-controller principle and stores CACAO and CSAF documents in a MongoDB. We provide a dashboard for all tasks and their states. Additionally, an analyst can manage the device representations and integrated sources. The pattern section is dedicated to the definition of matching terms.

Our evaluation is threefold. We first run the application, gathering the security advisories (input) to generate playbooks (output). Afterward, we manually assess the input and compare it with the output to assess the overall playbook quality and completeness. As input, we rely on security advisories from different sources for the respective devices. Therefore, we have integrated security advisories from three sources: Siemens ProductCERT, Cisco PSIRT, and CISA ICS CERT. Our application automatically fetches remediation advice from these sources and prevents us from fetching the same advisories multiple times. We selected these sources as they offer vendor-specific or aggregated security advice. Second, these sources ultimately use different data formats to evaluate whether structured security advisories lead to more qualitative and complete playbooks. We collected security advisories over the last 150 days for the playbook quality evaluation. As we also had to assess the security advisories manually, we considered only a collection period of 150 days, although our application could fetch and process even more advisories. After these advisories passed the whole process, we compare the following key indicators to evaluate the playbook's quality and completeness:

- Quantity of workflows actions
- Mistaken acceptance of workflow actions (*type I error*)
- Mistaken rejection of workflow actions (*type II error*)
- Classification of workflow actions

Third, we evaluate the performance of our automated process model showing that automation changes the game in managing vulnerabilities for ICS assets. For the performance measurement, we collect security advisories targeting our assets from the last five years. Through the manual labeling process, the human assessment and performance measurements, our experimental setting led to several results.

5.3 Experimental Results

We have grouped our results according to the process phases from security advisory to vulnerability playbook. Additionally, we show

²<https://www.eclipse.org/ditto/>

³A task manages the generation of one playbook.

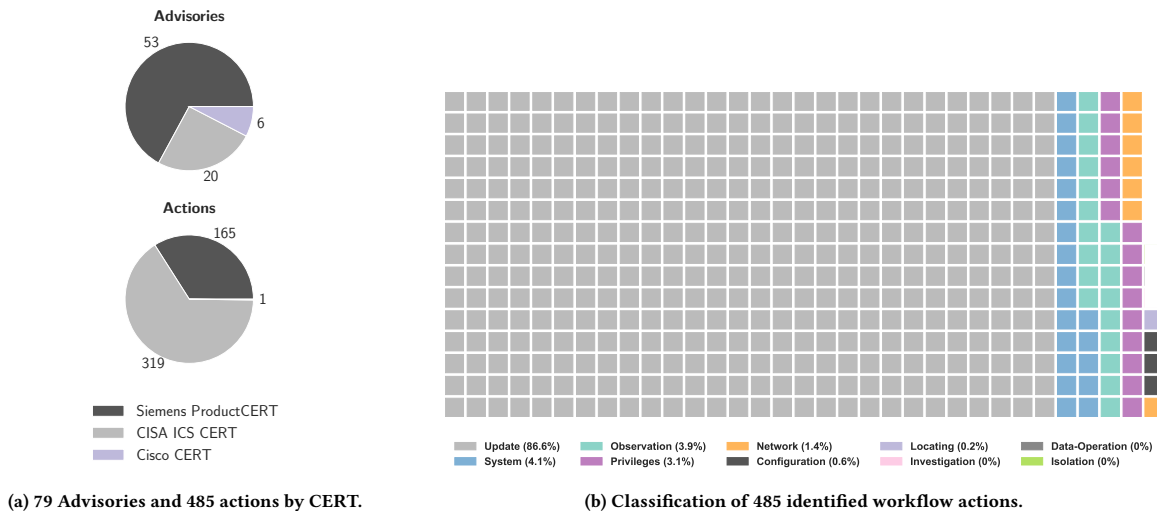


Figure 4: Analyzing the workflow actions in the generated CACAO playbooks.

results concerning playbook quality, completeness and performance. The results are documented using a Jupyter notebook to create transparency, which is available on GitHub⁴.

Vulnerability Search. In the industrial use case, device representations hold asset information, including CPE-IDs. We noticed that we could not assign a CPE-ID to each component. This problem has also been pointed out by previous research [3]. We found 13 CPE-IDs for the 17 Cisco assets and 22 CPE-IDs for the 20 Siemens assets. At first glance, these numbers sound reasonable but considering that CPE can address assets' firmware and hardware, we expected 34 and 40 CPE-IDs, respectively. In addition to the CPE-IDs, we added device-specific tags (i.e., model number). We found 35 vulnerabilities for our devices. Grasping the insecurity of ICS with these asset-specific vulnerabilities, we follow up with the search for security advisories.

Sourcing Strategy. Integrating the security advisory sources was a significant challenge due to their heterogeneity. The Siemens ProductCERT does not provide an API. Instead, they offer an Atom feed to query CSAF security advisories using the SSA ID⁵, CVE, title, product, sector, or tags. We use the advisory identifier within the Atom feed to manipulate the Siemens website URL and request the advisory in CSAF format. The Cisco PSIRT provides an API based on open security standards (e.g., CVE, CVSS, and CVRF)⁶. Since the API does only respond with XML-based CVRF, we still need to convert it. Finally, the CISA ICS CERT does not provide an API or feed to retrieve security advisories. Using the device tags of the device representations, we search within the HTML document and scrape information from its remediation section. As can be seen, searching for remediation advice without any interface

⁴<https://github.com/ad2play/evaluation>

⁵Siemens Security Advisory (SSA) is a Siemens global security advisory identifier.

⁶We noticed that the Cisco OpenVuln API recently added support for CSAF documents.

and filtering options is a fundamental problem. Therefore, we had to use the device tags and CPEs to automate filtering and verify whether the security advisory is associated with the asset and the respective vulnerability. Since they categorize vulnerabilities, products, and remediation steps, filtering is only a minor problem within CSAF/CVRF documents.

We identified 79 security advisories (see Figure 4a). Siemens offers 53 advisories, and Cisco offers six. CISA usually lists security advisories for both Siemens and Cisco devices, but the CISA advisories that have been fetched do not contain remediation advice for the Cisco device. However, Cisco has generally listed fewer advisories in the period in question. Also, CISA ICS CERT advisories primarily focus on ICS and do not cover Cisco products for IT enterprise networks. CISA provides a total of 20 security advisories for Siemens assets. It is also noticeable that Siemens offers several versions of advisories, but most overlap considerably in content. Therefore, the total number of Siemens advisories is significantly higher than those from CISA. In addition to the three sources mentioned above, we skimmed IBM X-Force Exchange and NVD for security advisories. There, we could find remediation advice only in linked external vendor documents creating complexity for our use case. At the end, we notice that different sources imply different obstacles in obtaining security advisories for specific assets, making sourcing inconvenient.

Advisory Conversion. After successfully acquiring security advisories, they are automatically converted into the CSAF data format. For Siemens advisories, already available in CSAF format, no further steps are necessary. The security advisories from Cisco and CISA are converted into CSAF using CVRF and HTML adapters, respectively. When the security advisories from all sources have been converted to CSAF, we analyze these documents.

A closer look at the remediation steps leading to workflow actions (see Figure 4a) also shows that the amount varies by vendor.

ARES 2023, August 29– September 01, 2023, Benevento, Italy

Anon.

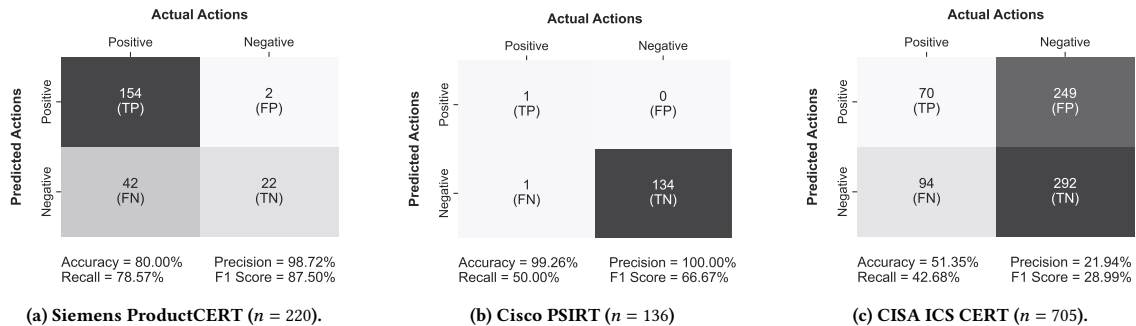


Figure 5: Measuring the CACAO playbook quality along workflow action classes (n equals the number of workflow actions).

While CISA has 319 remediation steps in 20 advisories (16 steps per advisory), Siemens captures 165 remediation steps (3 steps per advisory), and Cisco provides only one remediation step. The identified number of workflow steps in CISA might indicate a high type I error, but it is noticeable that CISA offers additional remediation advice compared to vendor-specific ones. Most interestingly, vendors even advertise remediation advice to inform customers that there is currently no fix available. None of the vendors directly offers technical commands (e.g., in OpenC2 or else) in the remediation steps, whereby dealing with textual descriptions of remediation advice is crucial. In conclusion, advisory conversion is strongly action-centric identifying individual remediation steps.

Playbook Generation. The standardized security advisories in CSAF enable the generation of CACAO playbooks. CACAO is an extensive standard and its implementation is challenging. Emblematic for this fact, the generated CACAO playbooks have a total length of 29,100 lines of code, which leads to 410 lines per playbook. However, generated CACAO playbooks are shorter than the initial CSAF documents. One reason is that CSAF also lists remediation advice for other assets, which were not required for our industrial use case. We successfully generated 71 CACAO playbooks out of 79 CSAF documents. Eight advisories require manual post-processing as actions could not be classified correctly. These eight reworks can be traced back to two issues. Seven errors are due to the NLP procedure, which has problems processing placeholders in version numbers, such as "update to version 3.X". The other error occurred because one remediation step could potentially be assigned to two different classes. Still, we can reduce the manual effort by roughly 90% and automating remediation advice can be seen as a success. Of course, final human assessment is crucial to performing the correct workflow actions to the right target at the right time.

Another elementary part of the CACAO playbook generation is the classification of the individual workflow actions (see Figure 4b). It is striking that 86.6% of the workflow actions force an update, whereas system (4.1%), observation (3.9%), and access (3.1%) play a rather subordinate role. We also found that the class observation is only mentioned in CISA security advisories. They have a dedicated section advising to observe malicious activity and to report security incidents. The lack of contextual understanding is also a problem while using NLP. These above numbers are the output we yield

within the automated process. Ensuring that the generated playbooks match the security advisories' content requires determining the overall CACAO playbook quality and completeness.

Playbook Quality and Completeness. We have already seen that the automated creation of CACAO playbooks is feasible and promising. We evaluate the extent to which these results are actually correct in the following. We measure the playbook quality and completeness by referring to confusion matrices (see Figure 5). These confusion matrices shows the three sources and an overall estimation of the playbooks' quality and completeness. We thereby include the correct amount of actions and their classification. We calculate the type I error as falsely identified workflow actions. The type II error represents the incorrectly rejected workflow actions. The total number of potential actions is given by the total number of sentences in the remediation steps ($= n$) because, except for one remediation step, all workflow actions were assigned unambiguously to a specific class. We assume that each workflow action is targeted by one sentence. Figure 5a shows Siemens security advisories' precision, accuracy, recall, and F1 score. The high precision (98.72%) shows a high quality of the generated CACAO playbooks. This indicates that the playbook quality is kept high when vendors provide security advisories in CSAF format. Only relevant remediation advice is included in the playbook generation process, while insignificant workflow actions are disregarded. The recall of 78.57% shows acceptable completeness of workflow actions indicating that only a small proportion of workflow actions is actually missing within the playbook. The CACAO playbook generation (F1 score = 87.5%) using structured and machine-readable security advisories is outstanding. The type I error is 1%, and the type II error is 19%, which signifies that the matching terms may be too soft. For example, the locating and isolation classes have been matched several times on the first dimension but did not succeed on dimension two. Figure 5b portrays the results for the Cisco PSIRT using the structured CSAF predecessor CVRF. This leads to an averaged result with an F1 score of 66.67%, a type I error of 0%, and a type II error of 0.007%. These results are insignificant, but we decided to include them for completeness. In contrast, unstructured security advisories from CISA deliver different stats (see Figure 5c). The generated playbooks for CISA are qualitatively inferior compared to Siemens, which is reflected by a low precision of 21.94%. The

Table 2: Performance of each process phase.

	Vulnerability Search	Sourcing Strategy	Advisory Conversion	Playbook Generation
∅	1.6 s/CVE	0.06 s/adv.	0.03 s/CSAF	0.06 s/CACAO
∑	7.42 min	27.23 s	14.38 s	20.74 s

identified workflow actions show higher incompleteness (precision = 42.68%), leading to an F1 score of 28.99%. The type I error is 35.3%, and the type II error is 13.3%. The direct comparison reveals that clear structured, machine-readable security advisories lead to more qualitative and complete playbooks, which in turn results in fewer manual corrections. The matching terms are an adjustment screw to balance the type I and type II errors, but the quality of the fetched security advisories is decisive.

Performance. We have found that fully automating the process, starting with vulnerability search and ending with playbook generation, saves time and reduces effort. For measuring the performance, we use the experimental setting mentioned above. We have collected vulnerabilities and security advisories for our devices for the last five years (as of December 2022). Table 2 shows each process phase’s average/total duration, respectively. It takes 7.42 minutes to lookup and filter vulnerabilities for 35 CPE-IDs and device tags (3784 unfiltered; 266 filtered). As components are not always mapped to a specific CPE-ID, our tool also performs searches with device tags. Due to the exhaustive filtering, we find long runs during vulnerability searches. Afterward, the tool uses this input to fetch 440 security advisories from different sources (Cisco: 112, Siemens: 267, CISA: 61), which takes 27.23 seconds (Cisco: 5.55s, Siemens: 11.08s, CISA: 10.6s). Siemens advisory sourcing takes twice as long because two different API calls are required; the first API call fetches the RSS feed, and the second downloads respective advisories. Advisory conversion takes 14.38 seconds (Cisco: 14.05, Siemens: 0s, CISA: 0.33s). As we use the dedicated Cisco API to transform CVRF to CSAF, these operations take longer. The automation successfully maps and generates 323 playbooks out of 440 advisories from these advisories in 20.74 seconds. In summary, using five years of historical data, it takes 8.46 minutes to automatically generate playbooks for our devices. We observe 1.57s on average to progress all process phases identifying a component’s vulnerability, deriving appropriate remediations, and generating a playbook. It is up to organizations to develop runtime (performance) optimization strategies and achieve higher scalability for complex environments.

5.4 Limitations

We have a few limitations concerning the application and evaluation. Design decisions had to be made in implementing our application following our process model. Therefore, we extended the JSON schema of CACAO and CSAF to a small extent due to the choice of specific technologies. For example, in the CACAO schema, we had to exclude trailing dollar signs for the data type identifiers to maintain compatibility with MongoDB. In addition, the proposed NLP procedure is inaccurate in terms of contextual understanding, the distinction between nouns and verbs, or sketchy texts. Our NLP implementation cannot accurately pinpoint the relationship

between actions and targets. Additionally, we can not identify the target. In addition, our evaluation has some further limitations. First, it is partway biased due to a large number of security advisories from Siemens ProductCERT and CISA ICS Cert. Hence, we can not generally argue about the generated playbooks’ quality and completeness across all security advisories. We can only observe that structured data yield better results than unstructured. Second, we have only connected three CERTs as potential sources for security advisories (limited to the last 150 days) based on our devices. And third, our playbook generation does not retain conditional logic or parallel flows (if existent in security advisories). The current mapping is rigidly sequential. Finally, we declare the handling of different versions of security advisories out of scope, e.g., those from Siemens ProductCERT.

6 RECOMMENDATIONS

We summarize the results and present recommendations for publishing security advisories directed at CERTs (*advisory publishers*) and automating ICS vulnerability handling directed at asset owners (*advisory subscribers*). The latter strongly depends on whether the publisher already provides ambitious remediation advice. Otherwise, subscribers have to assemble ambiguous remediation advice.

6.1 Publishing Security Advisories

We see a remarkable improvement potential for exchanging security advisories on the publishers’ side. Publishers (i.e., vendors and other CERTs) should enable more automated remediation advice retrieval for subscribers and foster a standardized exchange of security advisories.

Enable Automated Advice Retrieval. We have found that many data formats currently create a massive information overhead and expenses for subscribers of security advisories. One reason is that publishers only offer traditional communication channels, such as RSS feeds or email notifications. For a targeted query of relevant security advisories and to avoid information overhead, it is of utmost importance to offer a standardized API that additionally provides filtering options. APIs should leave it to the subscribers which data format they prefer for their remediation advice. This would make searching for security advisories less painful and more efficient.

Use Structured Security Advisories. Publishers should offer structured security advisories making the content easily machine-readable. Most data formats (i.e., HTML or PDF) for exchanging security advisories differ in structure and content. We have found that structured data formats (i.e., CSAF) better support automation than unstructured data by providing dedicated sections for actions and targets and tend to be more machine-readable. Translating unstructured data into machine-readable advisories requires sophisticated techniques coined by errors. In addition, structured data simplifies uniform handling without striving for different conversions of the security advisories. We also came across some best practices for the security advisories’ content. First, publishers should only include relevant information in security advisories to keep the remediation advice clean and to prevent information overhead. Second, publishers should be aware of streamlining, maintaining, and optimizing remediation advice. We believe versioning of security advisories

ARES 2023, August 29– September 01, 2023, Benevento, Italy

Anon.

to be helpful, as additional remediation advice extends to newly affected assets while keeping the total quantity of security advisories the same. Next, publishers should dedicate a sentence to each remediation step to foster automation. Additionally, as updates are not always feasible, publishers should include more “real” workarounds. Ideally, publishers should keep the CVE and product identifiers within security advisories. It can be observed that some security advisories do not list CVE-IDs. However, there is different remediation advice for different products and publishers should continue mapping product identifiers to individual remediation steps. If this mapping is missing, subscribers can not ensure that the remediation advice is meant for their assets. Last, we recommend publishing asset-specific commands, needed for automated playbook execution. For that purpose, CACAO already defines command types that can capture OpenC2 commands.

6.2 Automating Vulnerability Handling

Automating vulnerability handling is crucial to cope with the increased number of threats. We summarize our key learnings and provide recommendations for security advisory subscribers. Structured device representations, a clear sourcing strategy, the integration of machine learning, and the adoption of CACAO playbooks are enablers for automation.

Use Structured Device Representation. Subscribers must know their devices, components (including versions), and vulnerabilities. Comprehensive, well-structured, integrated device representations are the cornerstone for identifying and automating relevant remediation advice. We recommend using a structured format (e.g., JSON or Software Bill of Materials, SBOM) and device representations to model complex systems-of-systems. Enriching and maintaining these representations with security-relevant information (e.g., CPE) is crucial to identify vulnerabilities, exploits, and remediation advice.

Consider Sourcing Strategy. Subscribers should pay particular attention when selecting appropriate security advisory sources. As these sources differ in many aspects, subscribers have to decide whether the added value of a potential source outweighs the effort involved. The effort usually results from the additional development for security advisories’ conversion. For high quality, subscribers should directly integrate vendor-specific advisory sources if they plan automated processing. Free-to-use sources that aggregate remediation advice (e.g., CISA ICS CERT) list advisories from several vendors but are less suitable for automation. Alternatively, subscribers can obtain aggregated security advisories from security vendors without worrying about integrating different vendors.

Integrate Machine Learning. The integration of machine learning for the automated identification of actions and targets is promising. As long as some CERTs advertise remediation steps in plain text, subscribers should consider whether the application of machine learning can lead to a general improvement in automation. Sophisticated machine learning techniques could lead to sounder contextual understanding and, thus, better automation, quality, and completeness of workflow actions.

Adopt CACAO Playbooks. CACAO is a promising open standard. Subscribers should evaluate whether the CACAO standard eases maintaining the cybersecurity posture for their ICS. CACAO allows the definition of variables enabling a context-aware and asset-centric approach for quick and efficient remediation. For example, subscribers can define CACAO templates for action classes or even more specific operations, dynamically populate them with variables, and automatically generate context-aware playbooks. At the time of our research, security advisories are still premature, allowing only partial automation. However, implementing the CACAO standard is associated with great efforts. As long as there is no CACAO interpreter, subscribers must manually develop the CACAO playbook integration and execution. The main weak points of CACAO are the premature definitions of workflow actions, low adoption, and a small community.

7 CONCLUSION

Security advisories for ICS vulnerabilities include alternative remediation measures when simply updating to the newest version is not an option. We have generated ICS vulnerability playbooks using open CSAF and CACAO standards. Our approach is the first to combine the fields of security advisories and playbooks addressing organizations’ need to handle ICS vulnerabilities. While security advisories foster informing about vulnerabilities, playbooks are intended for workflow actions and eventually support automated execution. We have shown that crucial remediation advice can be included in CACAO playbooks by implementing a process model and experimenting with an industrial use case. ICS security advisories exist in various formats. Therefore, conversion to the CSAF standard is central to automated playbook generation. Towards the creation of individual workflow actions, we built upon matching terms to classify different remediation measures. In 79 security advisories, we identify a high prevalence of update advice and many less practical remediation steps. Our results lead us to recommendations for security advisory publishers and automated vulnerability handling. Improving security advisories’ structure and the content will help vulnerability playbook generation.

Future research can focus on further integration of open standards and their various features. While we use matching terms to extract workflow actions, machine learning techniques might be able to build technical commands and add conditional workflow logic. Towards automated playbook execution, we also see the necessity to incorporate organization-specific factors as remediation measures could be deliberately kept vague to serve all architectures and systems equally well. Our work is based on available ICS data. As a result, our vulnerability playbooks are specific to ICS. It is worth investigating vulnerability playbook generation for IT assets. Nevertheless, we see the two emerging open standards with increasing number of adopters shaping tomorrow’s security operations.

REFERENCES

- [1] Mehdi Akbari Gurabi, Avikarsha Mandal, Jan Popanda, Robert Rapp, and Stefan Decker. 2022. SASP: a Semantic web-based Approach for management of Shareable cybersecurity Playbooks. In *Proceedings of the 17th International Conference on Availability, Reliability and Security*. 1–8.
- [2] Luca Allodi, Sebastian Banescu, Henning Femmer, and Kristian Beckers. 2018. Identifying Relevant Information Cues for Vulnerability Assessment Using CVSS

- (CODASPY '18). Association for Computing Machinery, New York, NY, USA, 119–126. <https://doi.org/10.1145/3176258.3176340>
- [3] Anonymous Authors. 2020. Obfuscated title.
 - [4] Alexis Challande, Robin David, and Guénaél Renault. 2022. Building a Commit-Level Dataset of Real-World Vulnerabilities (CODASPY '22). Association for Computing Machinery, New York, NY, USA, 101–106. <https://doi.org/10.1145/3508398.3511495>
 - [5] Cybersecurity & Infrastructure Security Agency (CISA). n.d. ICS-CERT Advisories. <https://www.cisa.gov/uscert/ics/advisories> last accessed 2022-10-01.
 - [6] Marietheres Dietz and Günther Pernul. 2020. Digital Twin: Empowering Enterprises Towards a System-of-Systems Approach. *Business & Information Systems Engineering* 62, 2 (2020), 179–184. <https://doi.org/10.1007/s12599-019-00624-0>
 - [7] Philip Empl, Daniel Schlette, Daniel Zupfer, and Günther Pernul. 2022. SOAR4IoT: Securing IoT Assets with Digital Twins. In *Proceedings of the 17th International Conference on Availability, Reliability and Security*, 1–10.
 - [8] Stefan Fenz, Andreas Ekelhart, and Edgar Weippl. 2008. Fortification of IT security by automatic security advisory processing. In *22nd International Conference on Advanced Information Networking and Applications (aina 2008)*. IEEE, 575–582.
 - [9] Stefan Fenz, Andreas Ekelhart, and Edgar Weippl. 2008. Semantic potential of existing security advisory standards. In *Proceedings of the FIRST 2008 Conference-Forum of Incident Response and Security Teams*.
 - [10] Hugo Gascon, Bernd Grobauer, Thomas Schreck, Lukas Rist, Daniel Arp, and Konrad Rieck. 2017. Mining attributed graphs for threat intelligence. In *Proceedings of the Seventh ACM Conference on Data and Application Security and Privacy*. 15–22.
 - [11] Branden Ghena, William Beyer, Allen Hillaker, Jonathan Pevarnek, and J Alex Halderman. 2014. Green lights forever: Analyzing the security of traffic infrastructure. In *8th USENIX workshop on offensive technologies (WOOT 14)*.
 - [12] Chadni Islam, Muhammad Ali Babar, and Surya Nepal. 2019. A Multi-Vocal Review of Security Orchestration. *Comput. Surveys* 52, 2, Article 37 (2019), 45 pages. <https://doi.org/10.1145/3305268>
 - [13] Joseph R. Biden Jr. 2022. Executive Order on Improving the Nation's Cybersecurity. <https://www.whitehouse.gov/briefing-room/presidential-actions/2021/05/12/executive-order-on-improving-the-nations-cybersecurity/> last accessed 2022-10-01.
 - [14] Craig Lawson and Al Price. 2022. 2022 Market Guide for Security Orchestration, Automation and Response Solutions.
 - [15] Frank Li, Zakir Durumeric, Jakub Czyz, Mohammad Karami, Michael Bailey, Damon McCoy, Stefan Savage, and Vern Paxson. 2016. You've got vulnerability: Exploring effective vulnerability notifications. In *25th USENIX Security Symposium (USENIX Security 16)*. 1033–1050.
 - [16] Vasileios Mavroeidis, Pavel Eis, Martin Zadnik, Marco Caselli, and Bret Jordan. 2021. On the Integration of Course of Action Playbooks into Shareable Cyber Threat Intelligence. In *2021 IEEE International Conference on Big Data (Big Data)*. IEEE, 2104–2108.
 - [17] MITRE. 2022. Detection, Denial, and Disruption Framework Empowering Network Defense (D3FEND). <https://d3fend.mitre.org/> last accessed 2022-10-01.
 - [18] National Vulnerability Database (NVD). 2022. CVE-2022-34819 Detail. <https://nvd.nist.gov/vuln/detail/CVE-2022-34819> last accessed 2022-10-01.
 - [19] OASIS. 2019. Open Command and Control (OpenC2) Language Specification Version 1.0 - Committee Specification 02. <https://docs.oasis-open.org/openc2/oc2ls/v1.0/oc2ls-v1.0.html> Last accessed 2022-10-01.
 - [20] OASIS. 2021. CACAO Security Playbooks Version 1.0 - Committee Specification 02. <https://docs.oasis-open.org/cacao/security-playbooks/v1.0/security-playbooks-v1.0.html> last accessed 2022-10-01.
 - [21] OASIS. 2022. Common Security Advisory Framework Version 2.0 - Committee Specification 03. <https://docs.oasis-open.org/csaf/csaf/v2.0/csaf-v2.0.html> last accessed 2022-10-01.
 - [22] Forum of Incident Response and Security Teams (FIRST). 2022. Automation SIG. <https://www.first.org/global/sigs/automation/> last accessed 2022-10-01.
 - [23] ATC Project. 2020. RE&CT Framework Documentation. <https://atc-project.github.io/atc-react/> last accessed 2022-10-01.
 - [24] Daniel Schlette, Marco Caselli, and Günther Pernul. 2021. A Comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective. *IEEE Communications Surveys & Tutorials* 23, 4 (2021), 2525–2556. <https://doi.org/10.1109/COMST.2021.3117338>
 - [25] Saranyan Senthivel, Shrey Dhungana, Hyunguk Yoo, Irfan Ahmed, and Vassil Roussev. 2018. Denial of Engineering Operations Attacks in Industrial Control Systems. In *Proceedings of the Eighth ACM Conference on Data and Application Security and Privacy (Tempe, AZ, USA) (CODASPY '18)*. Association for Computing Machinery, New York, NY, USA, 319–329. <https://doi.org/10.1145/3176258.3176319>
 - [26] Avi Shaked, Yulia Cherdantseva, and Pete Burnap. 2022. Model-Based Incident Response Playbooks. In *Proceedings of the 17th International Conference on Availability, Reliability and Security*. 1–7.
 - [27] Florian Skopik, Giuseppe Settanni, and Roman Fiedler. 2016. A problem shared is a problem halved: A survey on the dimensions of collective cyber defense through security information sharing. *Computers & Security* 60 (2016), 154–176.
 - [28] Rock Stevens, Daniel Votipka, Josiah Dykstra, Fernando Tomlinson, Erin Quarataro, Colin Ahern, and Michelle L Mazurek. 2022. How Ready is Your Ready? Assessing the Usability of Incident Response Playbook Frameworks. In *CHI Conference on Human Factors in Computing Systems*. 1–18.
 - [29] Brandon Wang, Xiaoye Li, Leandro P de Aguiar, Daniel S Menasche, and Zubair Shafiq. 2017. Characterizing and modeling patching practices of industrial control systems. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 1, 1 (2017), 1–23.
 - [30] Moira J West-Brown, Don Stikvoort, Klaus-Peter Kossakowski, Georgia Killcrease, Robin Ruefle, and Mark Zajicek. 2003. *Handbook for Computer Security Incident Response Teams (CSIRTs)*. Technical Report. Carnegie Mellon University Software Engineering Institute.

6 Do you Play It by the Books? A Study on Incident Response Playbooks and Influencing Factors

Publication information

Current status:	Under review
Conference:	45th IEEE Symposium on Security and Privacy, San Francisco, United States of America, May 20 - 22, 2024
Date of submission:	14 April 2023
Full citation:	SCHLETTE, D., EMPL, P., CASELLI, M., SCHRECK, T., & PERNUL, G. (2024). Do you Play It by the Books? A Study on Incident Response Playbooks and Influencing Factors. Submitted to <i>The 45th IEEE Symposium on Security and Privacy (S&P 2024)</i> .
Authors' contributions:	Daniel Schlette 30%
	Philip Empl 30%
	Marco Caselli 20%
	Thomas Schreck 10%
	Günther Pernul 10%

Conference description: Since 1980, the IEEE Symposium on Security and Privacy has been the premier forum for presenting developments in computer security and electronic privacy, and for bringing together researchers and practitioners in the field. The 2024 Symposium will mark the 45th annual meeting of this flagship conference.

Do You Play It by the Books? A Study on Incident Response Playbooks and Influencing Factors

Abstract—Incident response “playbooks” are structured sets of operational procedures organizations use to instruct humans or machines on performing countermeasures against cybersecurity threats. These playbooks generally combine information about a given threat and organizational aspects relevant within the context of an organization. Both types of information are crucial for using, maintaining, and sharing playbooks across organizations as they ensure effectiveness and confidentiality. While practitioners show great interest in playbooks, their characteristics have not yet been thoroughly investigated from a research perspective. For this reason, we explore the topic by analyzing what is inside a playbook. Our approach consists of a comprehensive empirical assessment of available data (1221 playbooks), an online study with 147 participants, and final in-depth interviews with nine security professionals to consolidate and validate our findings. We notably find intrinsic ambiguities in the way practitioners and organizations define their playbooks. Furthermore, we notice that available playbooks cannot be used outright which might currently impair their wide use across different cybersecurity actors. As a result, we can conclude that organizations do “play it by the books” but individually define what is inside their playbooks and which areas of incident response they might address.

1. Introduction

Organizations facing cybersecurity threats should define and document standard operating procedures to ensure consistent cybersecurity operations and incident response [1], [2], [3], [4]. The shortage of skilled cybersecurity analysts [5], [6] draws additional attention to (automated) processes that can increase efficiency by reducing and eliminating errors. Integrating security tools, data, and teams is another pain point identified by cybersecurity practitioners [7], [8]. Dedicated processes can help organizations use resources effectively, particularly host and network tools, internal and external threat intelligence, and multiple security teams.

Playbooks represent processes and procedures which are part of every organization. In cybersecurity, more specifically in incident response and security operations, playbooks make for a novel field of research. While these playbooks address a specific threat or incident, they complement other well-established areas, such as system hardening [9], [10], vulnerability handling [11], [12], [13], [14], and business process management [15], [16]. System hardening, being mainly proactive, is based on checklists to fulfill generic security requirements (i.e., confidentiality, integrity, avail-

ability) during deployment. Vulnerability handling covers proactive scanning and security advisories where process representation (i.e., lists of remediations) is incomplete. Business process management and playbook-based IT operations (e.g., with Ansible [17]) show overlap but intentionally lack a clear cybersecurity focus. However, incident response playbooks address reactive scenarios, specifying what to do when things go wrong [18], [19].

With threats on the rise [20], there is a move toward playbooks as they promise consistency and automation. This development can be seen most clearly in the proliferation of Security Orchestration, Automation, and Response (SOAR) platforms [21], [22]. Playbooks are at the core of SOAR platforms that streamline security operations. Beyond vendors, standardization efforts and industry groups document the interest in playbooks and automation. The OASIS CACAO technical committee and the FIRST Automation special interest group aim to advance the representation of playbooks and explore user perspectives [23], [24]. As a result, playbooks are widely recognized, praised by professionals and playbook data has started to become available.

Nevertheless, open challenges around playbooks remain. Above all, we need to tackle the understanding of playbooks. Only a few previous works have discussed playbooks [18], [25], [26], [27]. Therefore, our work explores the overarching question: *What is inside a playbook?*

Initially, we observe the absence of an established theoretical playbook foundation. From an academic perspective, we suppose that playbooks are based on generic, technical information and contain additional organization-specific information. *Community playbooks*, describing generic operations and addressing existing threats, are stripped of any specific organizational context. They can be shared across different organizations as they do not convey confidential information. We expect to find these community playbooks analyzing repositories of different SOAR vendors.

When used within organizations, playbooks encompass additional organizational aspects defined by context. Making playbooks fit the context is about considering *influencing factors* and complementing community playbooks with organization-specific information. We expect organizations that define incident response processes and use playbooks to point out influencing factors to some extent. While incident response playbooks are a novel research field, our theoretical considerations are relevant for organizations using, maintaining, and sharing playbooks.

Against this backdrop, we aim to validate our playbook hypothesis by exploring the following research questions:

RQ1: *What are characteristics of community playbooks made available by trusted sources?*

RQ2: *Which influencing factors shape incident response processes and organization-specific playbooks?*

To answer these questions we gather, preprocess, and analyze playbook data from leading SOAR vendors (e.g., Splunk), cybersecurity institutions (e.g., CISA), and open-source platforms (e.g., Shuffle). In our three-step approach, we further perform an online study ($n = 147$) and conduct interviews ($n = 9$) to deepen the discussion on playbook content with specific attention to influencing factors. In particular, this paper makes the following contributions:

- We introduce a theoretical foundation to the playbook concept. Considering community playbooks and influencing factors specific to an organization is crucial when using, maintaining, and sharing playbooks.
- We are the first to empirically analyze 1221 playbooks from 14 sources. Our analysis reveals that community playbooks are modular and contain a significant amount of information about tools and ticketing which are likely handled differently across organizations.
- We find that incident response processes and playbooks are indeed shaped by influencing factors but this is not necessarily recognized by practitioners. Our online study shows that the number of steps and the workflow inside a playbook varies. Interview participants mention specific implications of influencing factors.
- We report that while playbooks are well-known, understanding and use depend on implementation tools and abstraction levels.

The remainder of this paper is structured as follows. In Section 2, we outline playbook foundations and use cases. We describe our methodology and introduce a research model for influencing factors in Section 3. We report on the characteristics of community playbooks in Section 4. We present results from our online study in Section 5. Interview findings are reported in Section 6. We discuss the implications in Section 7. Section 8 presents related work before we conclude our work in Section 9.






2. Playbook Foundations

We define the playbook concept and show basic forms of playbooks representation. Influencing factors must be considered when sharing, maintaining, and using playbooks in organizations as they transform community playbooks.

2.1. Playbook definition

Despite the growing interest, the notion of playbooks still needs to be clarified as other terms such as script, automation, runbook, checklist, standard operating procedure, workflow, or process are frequently used. Playbooks have been defined in a few other academic works [18], [28], [29], [30], [31], by cloud providers, and cybersecurity vendors. Most notable, according to the US President's

TABLE 1. TEXTUAL, GRAPHICAL, OR CODE-BASED ELEMENTS AND ASSOCIATED NOTATIONS CAN BE USED FOR PLAYBOOK REPRESENTATION

Type	Elements	Notation
Textual	Checklist  , Table 	Markdown
Graphical	Diagram 	BPMN
Code	File  , Script 	JSON, Python

Executive Order 14028, playbooks are “a standard set of operational procedures to be used in planning and conducting cybersecurity activity” [2]. Besides, standardization efforts (e.g., OASIS CACAO [23]) provide reliable information on playbooks and their components (see Appendix A for a detailed comparison of terminology and concepts).

In absence of an established incident response playbook definition, three characteristics guide our work: 1) *Security context*. Playbooks are used for cybersecurity purposes. 2) *Process representation*. Playbooks describe instructions to be followed. 3) *Technology integration*. Playbooks assist humans with technology and data. Upon these characteristics, a playbook to us describes a specific cybersecurity process or procedure based on a workflow with individual steps. Identified by [27], workflow steps are essentially action-artifact triplets describing who performs an action on an object. A playbook addresses a more or less specific threat or incident. We explicitly acknowledge different levels of abstraction found in playbooks.

2.2. Playbook representation

As playbooks represent processes and procedures, they logically structure information. Workflow steps can be structured sequentially, one after the other, or in parallel if the tasks and resources permit. Other conditions can define the subsequent workflow steps. A text-based community phishing playbook might consist of the following steps:

- **Start:** Phishing email received
 - Create ticket and assign incident handler
 - Analyze email header, content and attachments
 - Correlate threat intelligence (DNS logs, IPs, hashes)
 - Search and remove (unread) phishing emails
 - Update protection systems and notify users
- **End:** Incident closed

Who the playbook is for – humans or machines – determines the decision for either textual, graphical, or code-based representation. The phishing playbook above offers a generic human-readable description. Information is presented in text form with a checklist. Other playbook representations include graphical or code-based elements with different notations (see Table 1 and Appendix B, C). In the case of low-code/no-code SOAR platforms, human playbook designers construct graphical playbooks enriched with text and commands. The tools then generate and save a code-based representation for automation. As data formats and SOAR platforms enforce structured, codified representations we aim to analyze these community playbooks in this work.

2.3. Playbook sharing, maintenance, and use

Inspired by collaborative cybersecurity and Cyber Threat Intelligence (CTI) sharing [29], [32], [33], we assume a basic setting as illustrated in Figure 1. Organization A (producer) shares community playbooks with Organization B (consumer). We expect that any organization-specific information is excluded as organizations value confidentiality. If Organization A is a SOAR vendor, playbooks are kept generic to promote widespread adoption of its platform.

On the receiving end, Organization B intends to use the playbooks for automation, onboarding, or other purposes. The playbooks' instructions contain essential elements (e.g., actions) that apply across organizations. However, the playbooks lack contextual elements of incident response specific to Organization B. For instance, Organization B might decide to sinkhole DNS traffic instead of blocking via firewall. Also, Organization B might only have a few incident handlers or outsourced services constraining in-depth malware analyses. Therefore, Organization B and other playbook consumers must transform playbooks to fit their context. The knowledge of influencing factors is a prerequisite for transforming and using community playbooks.

A push toward collaboration and actionable advice amplifies the relevance of playbook sharing. Industry groups (e.g., Financial Services Information Sharing and Analysis Center, FS-ISAC [34]) and (national) coordination teams can provide their constituents with playbooks. Sharing playbooks will help a small organization build incident response processes from the ground up. A large conglomerate can use playbooks to improve existing processes. Sharing can also aid in establishing a language to discuss incident response.

Influencing factors are also crucial for playbook maintenance within an organization. As organizations evolve, the organizational context changes. Consequently, knowing about influencing factors and how they manifest helps to keep organization-specific playbooks effective. For example, an organization might extend its business operations to another country with a different jurisdiction. As a result, changes in influencing factors (e.g., laws and regulations) require the transformation of existing playbooks by adding further reporting steps. The same applies when technology is discarded as threats and playbooks become obsolete.

2.4. Playbook transformation

Using and maintaining playbooks relies on two separate inputs – community playbooks and influencing factors. Via transformation, these inputs lead to organization-specific playbooks. In this setup, the inputs can change:

- *Changing community playbooks* are the result of a changing threat landscape where new threats emerge. Community playbooks cover the various threats and attacks that demand countermeasures.
- *Changing influencing factors* are the result of new organizational characteristics and external requirements. Influencing factors have organization-specific values and implications.

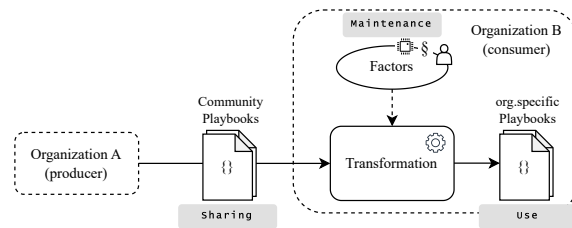


Figure 1. Sharing playbooks across organizations, maintenance, and use of playbooks put focus on influencing factors.

We build on these playbook foundations. What is inside a playbook is about analyzing community playbooks before examining how influencing factors shape organization-specific playbooks.

3. Methodology

We follow a mixed-methods approach combining quantitative and qualitative research to thoroughly investigate playbooks. Playbook data is analyzed to answer the first research question on community playbooks. With our online study and qualitative interviews we aim to answer the second research question on influencing factors. Data collection for community playbooks, online study, and interviews was carried out between Mai 2022 and February 2023.

3.1. Playbook data

We empirically analyze 1221 community playbooks from 14 sources for their characteristics. After collecting the data, we opted to automatically preprocess and analyze the playbooks, making the playbooks and analysis transparent¹.

Data collection. We initially tried to collect playbooks from all major SOAR vendors based on two popular market reports [21], [22]. Thus, we acquired access through GitHub repositories, engaged in a demonstration, or implemented the software through local deployment. However, we faced different obstacles. In particular, playbooks from several vendors were not publicly accessible, our requests remained unanswered, or vendors demanded non-disclosure agreements. To prevent any potential negative impact upon these vendors, explicit mention of their identities is deliberately withheld. To the best of our knowledge, we are the first to collect and compile a comprehensive playbook repository.

Data preprocessing. The data preprocessing was carried out using Python. We built custom connectors for each vendor to create schemata and extract relevant data from the heterogeneous playbooks. In addition, we used Jupyter notebooks to review and label 8,623 machine-generated actuator-action-artifact triplets.

1. <https://github.com/luduslibrum/awesome-playbooks> (anonymized)

Data analysis. Jupyter notebooks also describe our data analysis. We use NLP stemming, tokenization, and dependency parsing to extract the actuator-action-artifacts triplets based on the step name. We tested different clustering methods (e.g., k-Means, based on text vectorization and PCA) to identify step categories and also tried topic modeling methods. However, the methods remained unsuccessful mainly because the clusters or topics were ambiguous, leading us to manually label 370 unique actions to identify the clusters.

3.2. Research model and rationale

Extending the scope of our research to organization-specific playbooks, we set up a research model (see Figure 2). The model divides influencing factors in two groups – external and internal factors – and is used for our online study and interviews. Influenced by factors, incident response processes and playbooks are characterized by workflow logic and the content of individual steps. For groups and factors, we borrow from information security policy research [35], [36] and business process literature [37], [38].

We presented our idea of influencing factors at a major industry conference for incident response teams. When asked via interactive poll, all participants agreed that factors do influence incident response. However, answers started to vary when pressed, which is the most important factor. The feedback we received points to legal aspects, people, and technology. While organizations must determine if they can perform incident response processes as envisioned, they are also influenced by staff and available systems.

External factors. We assume attacker characteristics, including motivation, behavior, location, and others, to affect processes. Industry standards and guidelines, such as frameworks, can constitute an external factor as they serve as process references. In addition, laws and regulations require organizations to conduct incident response and implement compliant processes. These legal aspects capture, for example, business structure, location, and sector. We further assume that business partners and other organizations within the supply chain formulate expectations influencing organizational processes. Factor → impact examples are:

- Sophisticated attacker → long-term monitoring
- ISO/IEC 2700x → evidence collection
- EU GDPR → informing supervisory authorities
- Outsourced services → contact external team

Internal factors. Organizations can choose to abide by arbitrary internal rules or directives. Such internal factors can center on incident response data operations and attack targets. Besides, people form another internal factor which encompasses security culture and various security team-related elements (e.g., team size, chain of command, and responsibilities). The technology, general and cybersecurity specific, within an organization provides a setting in which incident response processes are placed. Thus, IT infrastructure and security tools might also explain variations in incident response processes. Factor → impact examples are:

- CRUD constraints → no data copying

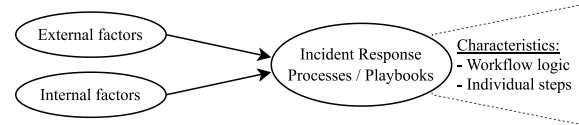


Figure 2. Theoretical research model with two factor groups influencing incident response processes and playbook use.

- CTI team → CTI sharing
- Security automation tools → querying endpoint agents

Observable differences. Discussing influencing factors directs attention to their observable influence on processes. We identify three possible changes caused by influencing factors – the first two center on workflow logic and the last on step semantics. We anticipate that incident response processes and playbooks can differ across organizations in the *number of steps*. Influencing factors might lead to more or fewer process steps changing workflow logic. Besides, we account for *sequential or parallel workflow*. Factors can introduce conditions to guide process flow, and process steps might be aligned sequentially or in parallel. At last, *altered steps* describe different actions. Influencing factors might affect the step’s action and thus alter semantics.

3.3. Online study

The primary objective of the online study is to assess the relevance of influencing factors within organizations. Therefore, we collected data from 147 participants reaching a broad audience within incident response. One of our co-authors with experience in online research methodology, provided expert guidance throughout our iterative process. Additionally, we incorporated feedback from a senior economics PhD student. To assess the validity and reliability of our questionnaire, we administered it to a group of colleagues in a controlled setting. In our questionnaire (available at our GitHub repository²) we first provide contextual cues pertaining to the authors’ affiliations and the scientific nature of the study, while also emphasizing the voluntarily participation. Furthermore, participants are instructed to consider their responses within the context of their respective organizations. Among other background questions, we directly ask for influencing factors (in three incident scenarios) and beliefs thereof. We have additional questions exclusively for participants using playbooks.

Participant recruitment and data. We recruit participants using our professional networks. We ask for participation via email, social media (i.e., LinkedIn), and industry groups (i.e., FIRST). To determine the specific source of each response, we created distinct questionnaires for each communication channel. Participants currently employed in cybersecurity are requested seven minutes of their time to partake. As it is difficult to get participants’ time, we provide

2. <https://github.com/luduslibrum/awesome-playbooks/tree/main/factors>

a monetary lottery incentive if they participate within the next 15 days. Facing unsolicited responses, we completely exclude any observations which contain invalid (i.e., non-English), redundant, and implausible answers. Exclusion is mainly based on free text fields (e.g., What is a playbook to you?) indicating missing security context (e.g., stage play) and numerical values. Further, redundant observations within a few-minute threshold containing only partial variations are excluded. In order to prevent the participation of bots, we included a math CAPTCHA as a security measure. We used the statistical software Stata to analyze the data.

3.4. Interviews

We interviewed security professionals working on incident response topics to gain deeper insights into influencing factors. While data saturation is arguable, we ceased the interview process after nine interviews as the results showed only marginal additions. The interviews were semi-structured to allow for flexibility and open-ended questions. Participants were not made aware of our research questions and instead asked about the effect of influencing factors in specific “what-if” scenarios. We opted for questions on incident response processes and procedures to remain unbiased of playbook notions. However, we actively asked additional questions whether and how participants use playbooks.

Interview guide. Following a pretest, the interview questions were adapted and reviewed by all authors. Our interview guide consists of four parts: 1) *Demographics*. We ask about a participant’s role and organizational characteristics. 2) *Incident response basics*. We ask about incident response within a participant’s organization. 3) *Incident response scenarios*. We outline three incidents (Ransomware, DDoS, APT) and ask about a participant’s organizational processes and implications of specific influencing factors. 4) *Playbooks*. We ask about playbook understanding and use.

Interview procedure. We conducted the interviews using cognitive interviewing methodology [39]. Interviews were held in German or English virtually via Zoom (invites sent by email), except for one in-person meeting. At the outset of the interviews, we introduced the topic, emphasizing the significance and relevance of the interview, and highlighting the research methods to collect and evaluate the data. Participants consented that notes were taken during the interviews. We further asked the participants if we could use their answers in anonymous or aggregated form for scientific publication, to which participants agreed. The interviews lasted from one hour to one hour and thirty minutes.

Participant recruitment. Participants working on incident response topics were selected based on personal contacts and the authors’ professional networks reflecting industry experience. Most interview participants work for large (> 5000 employees), multi-national corporations headquartered in Germany. Asked about how they would rate the maturity level for cybersecurity in their organization, most participants mentioned a high maturity compared to peers.

Coding and analysis. Interview questions and transcribed interview data are associated with influencing factors during data coding. We use thematic analysis [40], [41], [42] and corresponding deductive coding to relate questions to themes (see Appendix D). Our themes are based on the research model with its influencing factors (Section 3.2).

3.5. Ethics

Cybersecurity incidents and organizational processes are delicate topics. We validated our research approach with our institutions’ ethics committees via email explaining our research and data handling. We got confirmed that there is no legal obligation to obtain an ethics vote as very limited personal data is handled. While we were met with great openness, we aim to assure the anonymity of our study participants. Our questionnaire consisted of 43 questions, of which we left 37 open-ended. Participants gave their consent and had the option to opt-out at any point during the process. Notification was only provided to the winner of the lottery incentive, via email, while the rest remained unnoticed. We contacted potential interviewees twice, via email, after an initial contact attempt. We collected only a limited amount of demographic information, report mostly aggregated data, and in order to minimize errors, we allowed each interview participant to check and correct their quotes. We did not require non-disclosure agreements from any of our participants.

3.6. Limitations

While our study provides valuable insights, it is important to acknowledge its limitations, particularly with regard to the *playbook data*: The playbook data we analyze is mostly code-based, includes only a selection of vendors mentioned in two SOAR market reports, and is not representative of all platforms. This limits the scope of our analysis. Given the rapid pace of technological advancements and market consolidation, the community playbooks collected could become outdated fast. Another challenge we faced was the poor data quality in the community playbooks. The actuator-action-artifact triplets often contain typos, programming syntax, or incomplete data, making it almost impossible to analyze the data using clustering and topic modeling techniques. Although the step categories are highly specific, their contents can overlap due to blurry task boundaries (e.g., append or update data). The situation requires manual labeling of triplets, which can be prone to errors. We encountered ambiguity in labeling steps, often requiring us to assign them to multiple categories. In the *user study*, there could be potential biases and undetected duplicates related to the financial motivations of the participants, which may have influenced their responses. This may limit the reliability of our findings. Lastly, for the *interviews*, we faced challenges in identifying and accessing incident response professionals, making it difficult to obtain a representative sample.

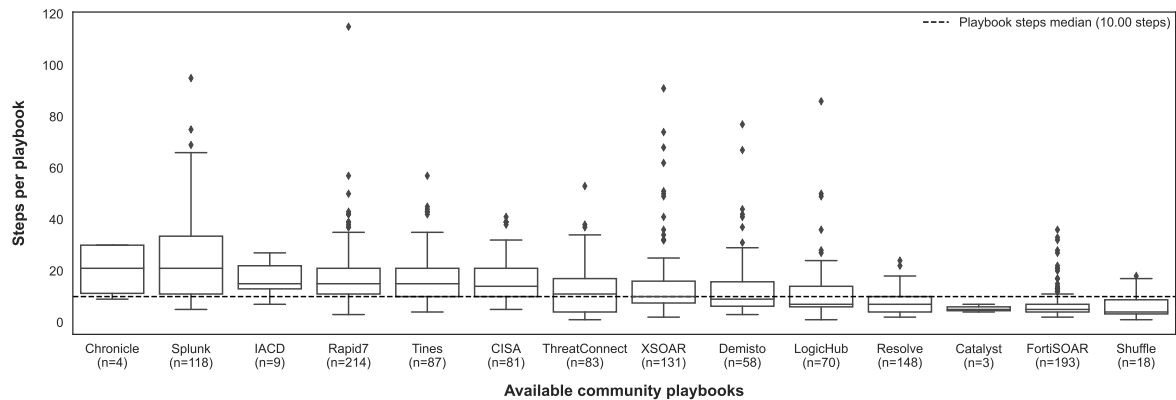


Figure 3. Exploring the characteristics of 1221 playbooks based on the number of steps.

4. Community Playbook Analysis

We present our findings of community playbook characteristics at playbook and workflow step level. Moreover, we categorize steps and show multi-category playbooks.

4.1. Playbook level

Playbooks exhibit high structural homogeneity across vendors, with variances predominantly found in their implementation-specific nuances. Below we summarize the components of an incident response playbook:

Meta information Descriptive elements, including an identifier, name, description, and categorization tags, serve to identify, retrieve, and understand a playbook. Playbooks also comprise specific details on attributes such as visibility, ownership, inheritance, or versioning.

Workflow At the core of a playbook lies the workflow. The workflow denotes a security process’s systematic and logical sequencing, characterized by a discrete set of individual steps discussed in more detail in Section 4.2.

Parameter Playbooks receive input and produce output. Inputs, outputs and other environmental parameters (i.e., playbook variables) can be defined globally at the playbook or workflow step levels.

Features As features, we define all playbook components that were used only occasionally. Such features encompass deployment specifications, playbook validation, testing procedures, process visualization, sharing policies, return on investment, or priority handling.

Community playbooks are mostly code-based and predominantly represented by JSON, YAML, and XML, with some sources opting for alternative tabular or graphical approaches in Business Process Modeling Notation (BPMN) [43]. There is a discernible inclination towards modularity to support community playbooks’ reusability. For example, some vendors rely on playbook hierarchies to allow “playbooks calling playbooks” and their unrestricted reusability

across individual use cases. These use cases leverage the logical linking of community playbooks. We find malware playbooks confirming this modularity. While one malware playbook is solely responsible for email notifications, the other handles malware hunting, and containment. We also find playbooks that include and test vendors’ platform integrations, e.g., testing the connection to Microsoft Teams.

4.2. Workflow step level

The workflow establishes a coherent sequence of steps. Each step comprises meta-information, including an identifier, a name, and a description. Moreover, a workflow step encapsulates the underlying logic, which guides the orchestration of the workflow. Notably, every workflow step contains an actuator-action-artifact triplet. However, we find that the relevant information can be conveyed implicitly.

Workflows differ significantly in their complexity and scope. Figure 3 compares community playbooks, indicating a median of 10 steps and a mean of 13.78 steps per playbook. However, the number of steps varies depending on the source, with 25% of the playbooks possessing less than six or more than 17 steps. Notably, a crucial aspect of SOAR is the automation of workflows, which is reflected in the degree of automation of the workflow steps. On average, 96.93% of the steps in the workflows are automated, whereas the remaining 3.07% of the steps necessitate human interaction. The rationale for human interaction stems from the workflow step type involved.

Workflow steps vary in their type. Based on 16,821 workflow steps, we identify seven types: start, end, single, decisions, loop, trigger, information, and playbook, confirming the CACAO [23] notion except for a parallel workflow step type. The start and end step types demarcate the scope of the playbook. More than half of the workflow steps are single, encompassing all process-relevant steps, including HTTP requests, filtering, and transformation tasks. Workflow steps orient on well-known process gateways, such as

TABLE 2. WORKFLOW STEP CATEGORIES, THEIR FREQUENCIES AND EXEMPLARY ACTUATORS, ACTIONS, AND ARTIFACTS.

Category	Frequency (abs./rel.)	Actuator	Action	Artifact
Logic	5175 (30.8%)	Platform, human, API	trigger, loop, decide	Playbook, while, condition
Utility	3748 (22.3%)	Platform, JMESPath, JoinArray	extract, filter, format	Status, data, report
Ticketing	2341 (13.9%)	Microsoft Teams, Slack, Jira	send, document, close	Email, ticket, case
Investigation	4622 (27.5%)	Human, VirusTotal, HashIt	link, search, lookup	Indicator (IP, domain/URL, file/hash)
Countermeasure	933 (5.5%)	AD, firewall, endpoint protection	block, quarantine, reset	Indicator, user, endpoint

decisions or loops. Decisions are made automatically via conditions or manually via ChatOps, prompts, and workflow interactions. Loops either wait for conditions to finish or conduct actions for each element in a list or array. Another type of workflow step comprises triggers that await a specific event. Informational workflow steps are also pertinent for informing humans, as they provide details about the current status or display relevant data. Lastly, certain workflow step types call other playbooks.

Accurately distinguishing between step types is challenging as some sources have integrated the process flow directly into each step. This means each step knows the predecessors and subsequent steps, including loops and conditions. Contrarily, other sources define the process flow globally in the community playbook, related to a node-link diagram. These varying approaches have far-reaching implications for the playbook's readability, length, number of steps, and step types. Another interesting aspect concerns how vendors integrate applications into their SOAR platform, including native integrations with their custom adapters or direct API calls to the application. Last but not least, from the total of 16,821 workflow steps, we identified 8,623 (51.3%) unique workflow steps indicating the reuse of steps within playbooks.

4.3. Categorization

We manually label unique workflow steps using the actuator, action, and artifact information and assign a specific category to a workflow step. We sometimes assign multiple step categories to a particular workflow step due to ambiguities. In these cases, a workflow step is related to multiple categories. Our findings on step categories are summarized in Table 2 and explained below.

Logic includes all the elements that play a crucial role in organizing and maintaining the process flow. The platform and the human are the primary actuators responsible for this category. For example, the security analyst decides the subsequent process path(s), calls additional playbooks, or initiates further workflow steps.

Utility encompasses all operations that are ancillary to the workflow's logic but crucial for supporting the other step categories. For instance, the platform utilizes regular expressions to extract data from a report.

Ticketing or alerting refers to the incident or case management to initiate a case and inform relevant stakeholders. A workflow step within this category may subsequently be reapplied to update a ticket.

Investigation deals with the search and analysis of relevant threat information, i.e., tactics, techniques, and procedures (TTP). A case in point would be a security analyst scrutinizing a phishing email, identifying an unfamiliar domain, examining its geographic location, and classifying it as an indicator of compromise.

Countermeasures go beyond analysis and take measures to counteract a threat through remediation, containment, and recovery. For instance, the security analyst mentioned in the previous example would block the IP address or domain of the phishing email.

Multi-category playbooks. As playbooks often comprise multiple step categories (e.g., logic steps are commonly integrated into workflows), it is imperative to investigate the composition of playbooks based on their step categories.

In Figure 4, we present the composition of multi-category playbooks and show the likelihood of encountering a particular step category within a playbook of n categories. To arrive at these findings, we analyzed playbooks containing at least two categories, accounting for 91.4% of the total playbooks. On average, playbooks mainly comprise logic and utility steps, with varying degrees of alerting/ticketing, investigation, and countermeasure steps. Besides, 7.2% of the playbooks feature all step categories, whereby the likelihood of encountering investigation steps within these playbooks is 34%. However, most playbooks have three different categories and the majority of these does not include countermeasure steps instead focusing on investigation (e.g., sighting inactive users or reviewing indicator reputations).

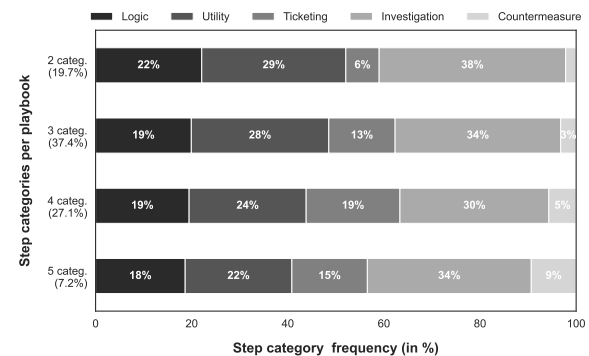


Figure 4. Multi-category playbooks and their workflow step composition.

TABLE 3. ONLINE STUDY PARTICIPANTS AND THEIR BELIEFS ON INFLUENCING FACTORS

Continent ($n = 144$)			
EMEA	84 (58.3%)	Asia	13 (9.1%)
America	47 (32.6%)		
Sector ($n = 141$)			
IT	74 (52.5%)	Industrials	13 (9.2%)
Public	21 (14.9%)	Consumer	7 (5%)
Financials	16 (11.3%)	Other	10 (7.1%)
Role ($n = 147$)			
Sen. sec. manager	58 (39.5%)	IT operations	11 (7.5%)
Sen. sec. expert	20 (13.4%)	Sec. researcher	11 (7.5%)
Sec. operations	15 (10.2%)	Sec. consultant	11 (7.5%)
Incident handler	15 (10.2%)	Other	6 (4.1%)
Influencing factors ($n = 142$)			
Technology	112 (78.9%)	Laws & regul.	74 (52.1%)
IR directives	98 (69%)	Attacker charact.	71 (50%)
People	94 (66.2%)	Supply chain	46 (32.4%)
Industry stand.	79 (55.6%)		

Additionally, we investigate whether playbooks consisting of five categories are intentionally designed for a specific incident type (e.g., malware or phishing). Upon analysis of 82 playbooks, we often find varying purposes, but also playbooks aligned to incident types. We derive that if a countermeasure step is included, there are likely ticketing and investigation steps in advance. This observation may account for the increased likelihood of countermeasures when more playbook categories are involved.

4.4. Implications for playbook adoption

Our results show that community playbooks are predominantly generic but contain organization-specific aspects. Workflow steps cover handling alerts and ticketing, which likely vary by organization. Meeting compliance requirements by adhering to service level agreements (SLAs) is present in community playbooks but must be determined in an organizational context. Further, community playbooks include people, technology, and best practices, but their adoption and use can vary significantly from organization to organization. For example, an organization may rely on a single analyst managing tickets or use divergent technologies, such as firewalls or communication tools, not integrated into vendor offerings. Ultimately, community playbooks are designed to match the vendors' product portfolios and focus on the vendors' customers as best as possible, not necessarily addressing the broad incident response community. Therefore, tailoring community playbooks to the organization's context is crucial to cater to the diverse needs (e.g., beyond automation) of different organizations. In the following sections, we will delve into the organizational context to better understand the factors that influence the transformation of community playbooks into organization-specific playbooks.

TABLE 4. ONLINE STUDY PARTICIPANTS SHOWING VARIANCE IN ORGANIZATIONAL PLAYBOOKS.

	Mean (Std.)	Median	n
Employees	35,756 (120,871)	600	145
Experience [years]	9.8 (6.5)	8	143
Security teams	2.9 (1.79)	3	142
IR team size	20.5 (36.2)	7	136
Team maturity [0-4]	2.7 (1.1)	3	145
Process maturity [0-4]	2.7 (0.9)	3	144
Tech. maturity [0-4]	2.7 (1.0)	3	144
Playbook contribution [0-4]	2.5 (1.25)	3	124
Org. playbooks	24.5 (46.8)	9	125
Malware steps	10.2 (7.9)	7	108
Phishing steps	8.4 (7.1)	6	102
Account comp. steps	8.6 (6.7)	9	100

5. Online Study Results

In this section, we report the findings of our online study, which included a sample of $n = 147$ participants. The data is presented in Table 3 and Table 4 for better clarity. First, we provide an overview of the participants' demographics and their respective organizations, including incident response capabilities. Subsequently, we investigate the prevalence of playbook sharing, use, and maintenance in the surveyed organizations. Additionally, we explore the differences between malware, phishing, and account compromise playbooks within these organizations, with participants indicating the number of steps required for each playbook. Finally, we detail how the participants perceive the significance of influencing factors.

Playbook understanding. The study primarily consists of participants from Europe, the Middle East, and Africa. Most participants are associated with the information technology (IT) sector, occupying senior positions like security managers, including SOC and CTI managers. On average, the surveyed organizations have three security teams in place, namely, incident response (CERT/CSIRT), threat intelligence, and security operations center (SOC). An incident response team, on average, comprises seven employees. All participants report that their organizations possess mature people, processes, and technologies for incident response. The term "playbook" is predominantly used by most participants, while some mention related terms like "runbook/workflow" (an automated playbook), "stories" (use cases), "standard operating procedure", or "incident management framework". This indicates a partial variance in the participants' perception of playbooks. The majority of respondents mention an active involvement in the playbook design process. On average, an organization has 24.5 playbooks, with a high standard deviation indicating stark differences.

Playbook use. According to the participants, the primary reasons for using playbooks are documentation (79.7%), automation (52.8%), compliance (51.2%), and onboarding (44.7%). The majority of playbooks are text-based (76.7%), followed by graphical (46.8%) or code-based (38.7%), which highlights the significance of documentation. In terms of operational use cases, playbooks are primarily utilized for

investigation/analysis (83.6%), countermeasures/mitigation (68.9%), and alerting/ticketing (60.7%). About 52.5% of respondents use playbook hierarchies also seen in community playbooks. Regarding the relevance of the aforementioned use cases, 40% of respondents indicate using external playbooks, all of which have been modified for organizational context. Moreover, 71.1% of respondents share playbooks internally, while 13.2% also share them externally.

Playbook differences. We analyze the differences between organization-specific incident-type processes or playbooks. Therefore, we asked the participants about their malware, phishing, and account compromise workflow characteristics: step count, parallel steps, and involved influencing factors. On average, 67% of respondents report that their incident-type playbooks include parallel steps. Conversely, this indicates that these playbooks can be logically distinguished from other playbooks. Regarding the number of steps, we observe a high standard deviation (\pm seven steps) in all kinds of playbooks. Malware playbooks contain the most steps on average compared to the other two, while the account compromise playbook has the highest median. All in all, we see a varying number of steps in the individual playbooks, which we attribute to the influence of factors.

Influencing factors vs. beliefs. Specifically, we asked participants to identify factors affecting incident response processes and whether their incident-type playbooks include steps that reflect these factors. The participants believe that technology is the most crucial factor, followed by incident response directives and people. This implies that technological advancements within an organization are seen to have an impact on incident response processes and playbooks. Incident response directives and the security team(s) also play a crucial role. In the case of malware incidents, technology remains the most significant factor, followed by incident response directives and attacker characteristics. Similarly, phishing playbooks are influenced most by technology, incident response directives, and the security team. Last, in the case of account compromise, attacker characteristics are stated as the most relevant factor, followed by technology and team. In all three scenarios, it is noticeable that the characteristics of the attacker are important, although the participants believe that they generally play a relatively minor role. This means that organizations strive to identify the attacker's motivation, particularly regarding account compromise. In general, we observe that different influencing factors shape incident response processes and playbooks.

To sum up, our observations indicate that organizations have specific playbooks indicated by step count, parallel workflow and factors. They commonly use them to document organizational security processes. Playbooks are also frequently shared and modified. We can not derive which factor influences playbooks the most as it strongly depends on the use case. However, technology, incident response directives, people, and attacker characteristics seem to be prevalent. It remains unanswered which influence a factor exerts on community playbooks. Therefore, our interviews with incident response professionals go into detail.

TABLE 5. INTERVIEW PARTICIPANTS ARE CHARACTERIZED BY INDUSTRY SECTOR, POSITION, AND ID.

Sector	Position	ID
Information Technology	Tech / Mgmt	P3, P6, P7
Industrials / Materials / Energy	Tech / Mgmt	P2, P4, P8, P9
Public Institution	Tech	P1, P5

Tech technical positions Mgmt management positions

6. Interview Findings

We describe for each external and internal factor the (missing) influence on incident response processes and playbooks. Table 5 presents details about the participants and their ID. Influencing factors and selected statements made by interview participants are listed in Table 6. The findings below depict aggregated data from the interviews.

Processes and playbooks are organization-specific. We had interview participants elaborate on their organization's incident response processes and playbooks. They offered detailed insights into what they do and what others might not be doing (e.g., investigations, CTI sharing, or coordination). Mostly, processes and playbooks are generic and aligned to incident types (e.g., phishing, malware, or account/system compromise) or recurring events. We can report that all interview participants were aware of playbooks or used playbooks as *“they can reduce time spent on tasks, improve task quality, and allow rookie teams to have stable operations.”* (P8) Whether processes and playbooks are organization-specific, participant P3 sums up a common stance: *“The incident response team creates playbooks [...]. [Community playbooks] never fit the organizational context and are either too generic or too specific for certain products.”* (P3) In combination with other participants' statements this leads us to conclude that organizational context and factors matter.

Attacker behavior and motivation beat location. Behind every attack or incident, there is a threat actor. Organizations fending off attacks and coping with incidents take attacker characteristics into account. However, interview participants paint a nuanced picture when asked about the influence of more specific attacker characteristics, such as presumed location, behavior, and motivation, on their processes. Unintuitively, precise attribution, whether attacks originate from a specific group and country, is considered less relevant as *“clarifying attack origin is clearly not the duty of private corporations but should be delegated to law enforcement.”* (P2)

As explained by participants' narrow understanding of attribution as solely identifying the attacker's location, *“it is all about technical matters rather than where the attack originated from.”* (P6) Consequently, the focus is shifted to the influence of attacker behavior and motivations on processes. Beyond prioritization, security teams escalate to management and conduct additional threat hunting in an Advanced Persistent Threat (APT) scenario. Different motivations (e.g., espionage, financial gain, or reputation)

TABLE 6. INFLUENCING FACTORS AND STATEMENTS MADE BY INTERVIEWED PROFESSIONALS.

	Influencing Factor	Participant statement
External	Attacker characteristics	“We investigate for longer when we think there is more to know [in the case of an APT].” (P4)
	Industry standards	“We started with ISO 27035 and developed a [high-level] process according to our organizational needs.” (P8)
	Laws and regulations	“Playbooks and actions are pre-approved by legal. We keep lawyers in the loop [during incident response].” (P7)
	Business structure	“We have a general [workflow] step to check if authorities need to be informed.” (P4)
	Location	“We perform EU GDPR assessment and inform authorities.” (P2)
	Sector	“Only incident handlers in the US are allowed to handle incidents affecting US military contracts.” (P3)
	Supply chain	“We collaborate with an ISP [for DoS attacks].” (P1)
Internal	Incident response directives	“Playbooks go into detail. A step refers to ‘investigate headers’ and which specific tools to be used.” (P7)
	Data operations	“We manually extend data retention adapting the log settings [in the case of an incident].” (P1)
	Targets	“We have a crisis mode, a dedicated crisis team, and retainers with [...] forensic companies.” (P3)
	People	“We do shift operations, on-call duty, and follow the sun for incident response.” (P9)
	Security culture	“Permissions and pre-authorizations are important. At times you have to act without [explicit] permission.” (P2)
	Security team	“We diligently share information, insert data in MISIP, and correlate incidents.” (P4)
	Technology	“Business units or third parties operate our infrastructure. We advise and coordinate measures.” (P6)
	IT infrastructure	“If necessary, we integrate third parties (AWS, Microsoft) into our process. They are notified by email.” (P9)
	Security tools	“Without a [SIEM/SOAR] dashboard, we rely on [administration] tools to detect and respond to incidents.” (P5)

are reflected in decisions to observe or eliminate threats: “We investigate for longer when we think there is more to know [in the case of an APT].” (P4)

Industry standards provide broad guidance only. Organizations build on industry standards and guidelines for their high-level incident response process. Overarching frameworks (e.g., NIST Cybersecurity Framework, Incident Response Life Cycle, or FIRST CSIRT framework) are often adapted: “We started with ISO 27035 and developed a [high-level] process according to our organizational needs.” (P8) On a more detailed level addressed by playbooks and automation scripts, we cannot infer any influence of industry standards and guidelines from the interviews.

Laws and regulations apply throughout. A central finding in our validation was the relevance of legal aspects for incident response. In our interviews, participant P7 summarized the influence of laws and regulations on processes at different levels: “Playbooks and actions are pre-approved by [our] legal [department]. We keep lawyers in the loop [during incident response].” (P7) We conclude that any initial process design and playbook transformation must consider legal influences. Defined roles and communication between security and legal teams can help streamline security processes, especially in unfamiliar situations.

When examining the influence of more specific legal aspects, we make the following observations:

- Business structure requires compliance, but the influence is vague: “We have a general [workflow] step to check if authorities need to be informed.” (P4)
- Location-based influence is mainly about privacy and data protection: “We perform EU GDPR assessment and inform authorities.” (P2)
- Sector-based influence, most notable in the defense industry, has implications on security operations: “Only incident handlers in the US are allowed to handle incidents affecting US military contracts. We do reassign responsibilities accordingly.” (P3)

Overall, there is a strong emphasis among participants that technical incident response is separate from legal processes. For instance, while we expected to hear about SEC 8-K filings for NYSE-listed corporations or regional equivalent regulations, our participants from large organizations instead delegated responsibilities. Nevertheless, participants indicated the relevance of organizational aspects in their processes: “Our legal department gets [corresponding] tickets in their ticketing system when [a GDPR-related] incident happens.” (P7) Beyond Europe, a legal influence was mentioned for the USA but not for other countries: “If the US is concerned, we have separate procedures.” (P4)

When in doubt, collaborate with business partners. Recent attacks and proposed legislation (e.g., EU Cyber Resilience Act) put focus on cybersecurity within the supply chain. However, we do not see a clear indication of the influence of business partners’ expectations on specific processes within the interview results. We note that most participants work for larger organizations which, whenever necessary, collaborate with other organizations to resolve incidents: “We collaborate with an [Internet Service Provider,] ISP [for DoS attacks].” (P1) Another explanation for missing influence is the overlap with other factors, such as laws and regulations or internal technology sourcing strategies.

Besides external influencing factors, we evaluate internal factors shaping incident response processes and playbooks.

Directives address the finer points of incident response. Organizations have flexibility in how to handle incidents. Thus, individual workflow steps and actions can contain precise instructions permitting or constraining specific data operations (e.g., data copying) and addressing specific attack targets (e.g., devices of board members). Our interview participants mention SLAs that define incident response. For instance, when to start working on a case or report to management, but there is no definite time to close a case. On a technical level, tool selection is covered: “Playbooks go into detail. A [workflow] step [within a phishing playbook]

refers to ‘investigate headers’ and which specific tools to be used.” (P7)

Beyond these observations, we find the following:

- When organizations have to deal with storage constraints they perform additional data operations: “We manually extend data retention adapting the log settings [in the case of an incident].” (P1) This is partially contrasted by large organizations that have nearly “unlimited” storage capacity: “We aim for storing logs for a year. The more data, the better.” (P4) Nevertheless, costs are considered.
- Organizations adapt incident response when attack targets are business critical (e.g., systems or accounts). We note that these situations require a special process, are handled with priority, or are taken “off-process”. As a result, precautions are taken: “We have a crisis mode, a dedicated crisis team, and retainers with incident response and forensic companies.” (P3) Additionally, access to information might be restricted. Some participants object that incident response is always on a need-to-know basis, but they agree that different people (e.g., CISO) must be informed.

More people imply more tasks and coordination. Depending on the organization, one or more security teams are involved in incident response. Most participants state that they have incident response and other teams (e.g., IT operations, red team, threat intelligence, human resources) work together. Communication and the chain-of-command are relevant prerequisites as organizations “do shift operations, on-call duty, and follow the sun for incident response.” (P9) In favor of fast operations, organizations opt for specialization so tasks can be shared and executed in parallel. Security culture and teams influence incident response processes:

- Security culture is about management support. Influence on communication and permissions can be observed: “Permissions and pre-authorizations are important. At times you have to act without [explicit] permission.” (P2) In general, participants mention good management support within their organization but those without face consequences (e.g., missing resources), which affect tasks. As a result, tasks cannot be performed, and security becomes a topic of personal motivation: “Within a public institution, processes take time, and focus is on essential tasks.” (P5)
- In multi-national organizations when specialists and different teams work on incidents, additional tasks cover forensic investigations, dark web searches and CTI sharing: “We diligently share information, insert data in MISP, and correlate incidents.” (P4) Among the interview participants we notice different operation modes such as double incident assignments, task rotation, or feedback loops and different team structures. While in one organization certain tasks are performed by the incident response team itself, in other organizations these are handled by CTI, SOC/monitoring, or other dedicated teams: “We need to involve the email team [in the case of account compromise].” (P3)

On the shoulders of technology. Technology is a broad category influencing incident response processes and playbooks in various ways. Consequences range from general applicability (i.e., a process or playbook is relevant to the organization) to which specific actions can be taken (e.g., query EDR agents on every endpoint). Overall, influence is dependent on existing technology and its characteristics (e.g., centralized vs. decentralized, homogeneous vs. heterogeneous). Some participants point out that they only advise and coordinate actions because they do not (directly) operate the IT infrastructure: “Business units or third parties operate our infrastructure. We advise and coordinate measures.” (P6) Technology being used to fulfill business and security requirements has the following implications:

- IT infrastructure and sourcing strategies play an essential role. Nowadays, web services are typically hosted externally: “If necessary, we will integrate third parties (AWS, Microsoft) into our process. They will be notified by email but not via incident tracking.” (P9) Participants further mention that organizations can realize more streamlined processes if they have all their services in the cloud using one cloud provider. In contrast, OT networks and production systems are more diverse, demanding different measures.
- Organizations strive for redundant and independent incident response infrastructure: “Semi-self-sufficient operation is possible. We cover cases when no domain controller is available. The CERT should still be functioning and able to access their workstations.” (P8) This implies that processes cover switching to alternative infrastructure which includes communication channels (e.g., email, Mattermost, or war rooms). In addition, security tools shape what is done. Although most participants use commercial SOAR platforms, some have different solutions: “Without a [SIEM/SOAR] dashboard, we rely on [administration] tools to detect and respond to incidents.” (P5)

7. Discussion

We discuss implications resulting from our research on community playbooks and influencing factors. We emphasize how to cope with terminology, playbook content and handling playbooks.

7.1. How to talk about playbooks

Although organizations use playbooks, *what is a playbook to you?* must remain a key question due to ambiguous definitions. Toward a detailed understanding of playbooks offered by a given community or used within a given organizational context, our analysis recommends the following:

- Clarify playbook representation and implementation.
- Clarify abstraction levels and management instruments.

When it comes to community playbooks, it is best to pay attention whether or not they are code-based and linked to a SOAR platform. Emphasized by our community playbook

analysis, these playbooks typically rely on additional plugins (e.g., Docker containers, APIs, or other software artifacts) offered by the SOAR platform integrating (external) services and functions. Therefore, community playbooks abstract technical measures to some extent and largely do not contain specific CLI commands or code snippets. Community playbooks need to be checked for technical content and overarching concepts (e.g., use cases) to grasp the meaning of the term playbook.

When it comes to organization-specific playbooks, it is necessary to account for other organizational management instruments. Thereby, policies, plans, checklists, and security tools can define different abstraction levels and guide processes. Broadly speaking, playbooks can tilt toward text-based instructions within a knowledge base (e.g., wiki) or toward dedicated automation scripts both being possibly referred to as playbooks. Thus, it is necessary to inquire management instruments and tools. Additionally, a separation in response and detection playbooks should be checked.

7.2. How to define playbook information

Playbooks should be precise and contain information to fulfill their intended purpose. Analyzing community playbooks and manually labeling workflow steps, we discover that basic information on who is performing an action on an object (i.e., actuator, action, artifact), as mentioned in [27], is difficult to extract, obstructing comparison and perhaps use. Thus, we see potential for standardization, including naming conventions. While we initially agreed with research findings in [18] that (executable) commands are required inside a playbook, this is opposite to how organizations see their playbooks and community playbooks look like. Instead, most organizations aim for abstract playbooks and introduce another tool-based implementation layer below.

Nevertheless, organizations show intrinsic ambiguities in the way they define their playbooks. Even without clear statistical significance, we believe that there is merit in discussing influencing factors on incident response processes and playbooks. As community playbooks cannot be used outright due to missing or obsolete information (e.g., logic, tools, ticketing), transformation according to organizational context and consideration of influencing factors is needed. Providing an initially stepping stone for theory building, we want to emphasize three insights.

First, there is no clear picture on influencing factors. In our online study, most participants believe technology exerts influence, and they have steps based on (security) technology. In our interviews, participants show a deep knowledge but are not fully aware of what shapes their playbooks, possibly explaining the absence of some factors. Despite multiple participants claiming that incident response is solely about technical matters, their responses indicate otherwise. In line with other works on security management [44], [45], we argue that interfaces to other teams and technology are the most important factors. Acknowledging challenges on boundaries, organizations should define the scope of incident response and its playbooks.

Second, as community playbooks partially contain organization-specific information, we hypothesize that organizations prefer adapting available playbooks to building them from scratch. In [18], playbook frameworks guide playbook design. Intending to make intuitive influencing factors visible, we showcase an additional path to playbook design, assisting organizations in adapting community playbooks. Consequently, this might reduce the time and resources required to build organization-specific playbooks while including all essential elements.

Third, what should be inside an organizational playbook must be squared with organizations' intend. Organizations might opt for creativity and critical thinking of their incident handlers, avoiding specifics. In contrast to behavioral research emphasizing bias and unwanted variability (noise) [46], [47], [48], we reason that some threats (APTs) demand flexibility and that more specific playbooks could introduce further challenges (e.g., prioritization, reusability, hierarchies). Nevertheless, playbooks can detail when and which decision to make, thus building a cornerstone to combat bias, ensure consistency, and speed up repetitive tasks.

7.3. How to use, maintain, and share playbooks

For organizations using playbooks, the next step involves automation, tool selection, and deployment. However, it is crucial to determine if automation does help or harm. Interview participants mention that security professionals must be kept in the loop when making decisions. Besides automation and seen in the online study, playbooks can also aid documentation, reporting, and onboarding thereby improving existing processes. Ideally, building organization-specific playbooks is based on systematically merging community playbooks and influencing factors. Here, it is on future research to address recent developments of generative artificial intelligence for incident response (e.g., Microsoft Security Copilot's promptbook), investigating options to build highly contextual playbooks automatically. For playbook maintenance, organizations need to recognize and keep track of changes. Influencing factors could be kept in a dedicated repository and frequently checked. With separated repositories, organizations can build their organization-specific playbooks on-the-fly coping with a growing number of threats and playbooks over time. Playbook sharing is the foundation toward establishing a common language for incident response. We see challenges in diverse or new data formats and different sharing modes (e.g., internal recipients wish for commands) to be addressed in the future.

8. Related Work

The idea of structured guidance achieved with playbooks is present in adjacent research areas. We first mention related work on incident response playbooks before discussing system hardening, vulnerability handling, IT operations, and business processes on which playbooks are based.

8.1. Incident response playbooks

Incident response playbooks caught researchers' interest. Closest to our work, Stevens et al. [18] investigated playbook design pointing to organizational constraints. Schlette et al. [27] compared structured playbook representations building a prerequisite to playbook sharing. In [26] the removal of confidential information is briefly discussed. Similar to how playbooks provide strategic guidance in sports, their use in fighting DDoS attacks and establishing incident response programs centers on outlining different options [19], [25]. In cybersecurity, playbooks can be seen as a continuation of threat intelligence research aiming to present and disseminate actionable security information [32], [49], [50]. We find security standardization efforts encompassing playbooks and courses of action [29], [31], [51]. Aside from academic research, two GitHub repositories aggregate information on incident response [52], [53].

8.2. Structured guidance and systematic processes

Organizations perform system hardening to reduce their attack surface with securely configured systems [9], [10]. System hardening and security configuration rely on best practices and their technical implementation relates to playbooks. What must be done to conform with a given security baseline is defined by vendors (e.g., Microsoft security baselines [54]), community organizations (e.g., Center for Internet Security, CIS Benchmarks [55]), and governmental institutions (e.g., US Department of Defense Security Technical Implementation Guides, DoD STIGs [56]). Typically, as part of the Security Content Automation Protocol (SCAP), XCCDF-structured security checklists are used. They contain rules, fixes and can link to OVAL-based vulnerability checks. Implementation is based on tools (e.g., Ansible, Chef, Terraform), dedicated scripts (e.g., PowerShell, bash), and other configuration options (e.g., GPO backups). We notice similarities (e.g., abstraction, tools) and differences (e.g., proactive) compared to incident response playbooks.

Once vulnerabilities have been discovered and disclosed [14], organizational vulnerability handling involves preventing exploitation by following security advisories and patching systems [11], [12], [13], [57]. Typically, organizations are guided by CVRF/CSAF-structured security advisories describing how to fix a vulnerability [58]. We note that incident response playbooks focus on organizational processes and thus go beyond the rich stream of vulnerability research.

Extending the scope to non-security areas, IT operations, system administration, and configuration management cover systematic deployment, maintenance, and monitoring [59], [60]. Consequently, regular tasks are structured and automated with scripts or runbooks to ensure consistency [61]. Automation relies heavily on tools, with Ansible and its playbooks being a prominent example [17], [62]. In cloud environments, playbook-like concepts are part of *Infrastructure as Code* [63] but also remain security agnostic.

8.3. Business process orchestration

Process orchestration and automation are tightly coupled with SOAR platforms but are rooted in business process research [15], [64]. Business processes and process orchestration are necessary when there are multiple activities and diversity in people, processes, and technology [65], [66]. Business Process Management Systems (BPMS) define execution engines and perform fundamental tasks, such as modeling, simulating, instantiating, and monitoring workflows (i.e., automated business processes) [67], [68]. Besides, business process re-engineering relates to influencing factors and targets "order-of-magnitude improvements" [37], [38], [69]. Dedicated software solutions (e.g., Camunda, ServiceNow) implement execution engines and automation platforms for business processes. Comparing architectures and functionalities, we note that SOAR platforms and security orchestration largely borrow from business process research but instantiate concepts within the security context. A notable difference is the deep integration of security tools.

9. Conclusion

In brief, we investigated different factors that influence and shape the sharing, maintenance, and use of incident response playbooks. Our initial belief that playbooks have a clear and concise definition was challenged as we discovered that individuals have different understandings of what constitutes a playbook. Besides, organizations intentionally keep playbook information abstract to support flexibility and reuse, which is partially reflected in community playbooks. Although our research does not statistically clarify the factors' relevance, it makes factors visible, guiding playbook design and use. Our findings suggest that technology and the security team(s) are critical drivers in shaping incident response playbooks. For future work, we see opportunities to explore the systematic merging of community playbooks and influencing factors to build organization-specific playbooks. The emergence of generative artificial intelligence has already begun to reshape incident response processes, and we are keen to observe how playbooks and their use will continue to evolve in future research.

References

- [1] ISO/IEC, "ISO/IEC 27001:2022 - Information security, cybersecurity and privacy protection — Information security management systems — Requirements," International Organization for Standardization, Tech. Rep., 2022.
- [2] Executive Office of the President, "Executive Order 14028 of May 12, 2021 – Improving the Nation's Cybersecurity," 2021, last accessed 2023-03-01. [Online]. Available: <https://www.whitehouse.gov/briefing-room/presidential-actions/2021/05/12/executive-order-on-improving-the-nations-cybersecurity/>
- [3] P. Cichonski, T. Millar, T. Grance, K. Scarfone *et al.*, "Computer Security Incident Handling Guide," National Institute of Standards and Technology (NIST), Tech. Rep. SP 800-61, Rev2, 2012.
- [4] M. J. West-Brown, D. Stikvoort, K.-P. Kossakowski, G. Killcrece, and R. Ruefle, "Handbook for Computer Security Incident Response Teams (CSIRTs)," Carnegie-Mellon Software Engineering Institute, Tech. Rep., 2003.
- [5] (ISC)², "(ISC)² Cybersecurity Workforce Study 2022," (ISC)², Tech. Rep., 2022.
- [6] J. Dykstra and C. L. Paul, "Cyber Operations Stress Survey (COSS): Studying fatigue, frustration, and cognitive workload in cybersecurity operations," in *11th USENIX Workshop on Cyber Security Experimentation and Test (CSET 18)*, 2018.
- [7] R. Brown and P. Stirparo, "SANS 2022 Cyber Threat Intelligence Survey," SANS, Tech. Rep., 2022.
- [8] C. Crowley and B. Filkins, "SANS 2022 SOC Survey," SANS, Tech. Rep., 2022.
- [9] K. Scarfone, W. Jansen, and T. Miles, "Guide to General Server Security," National Institute of Standards and Technology (NIST), Tech. Rep. SP 800-123, 2008.
- [10] J. Christensen, I. M. Anghel, R. Taglang, M. Chiroiu, and R. Sion, "DECAF: Automatic, Adaptive De-bloating and Hardening of COTS Firmware," in *29th USENIX Security Symposium, USENIX Security 2020, August 12-14, 2020, 2020*, pp. 1713–1730.
- [11] S. de Smale, R. van Dijk, X. Bouwman, J. van der Ham, and M. van Eeten, "No One Drinks From the Firehose: How Organizations Filter and Prioritize Vulnerability Information," in *2023 IEEE Symposium on Security and Privacy, SP 2023, Proceedings, 22-25 May, 2023, San Francisco, California, USA, 2023*.
- [12] N. Alomar, P. Wijesekera, E. Qiu, and S. Egelman, "'You've Got Your Nice List of Bugs, Now What?' Vulnerability Discovery and Management Processes in the Wild," in *Sixteenth Symposium on Usable Privacy and Security, SOUPS 2020, August 7-11, 2020, 2020*, pp. 319–339.
- [13] K. A. Farris, A. Shah, G. Cybenko, R. Ganesan, and S. Jajodia, "VULCON: A System for Vulnerability Prioritization, Mitigation, and Management," *ACM Transactions on Privacy and Security (TOPS)*, vol. 21, no. 4, pp. 16:1–16:28, 2018.
- [14] D. Votipka, R. Stevens, E. M. Redmiles, J. Hu, and M. L. Mazurek, "Hackers vs. Testers: A Comparison of Software Vulnerability Discovery Processes," in *2018 IEEE Symposium on Security and Privacy, SP 2018, Proceedings, 21-23 May 2018, San Francisco, California, USA, 2018*, pp. 374–391.
- [15] M. Dumas, M. L. Rosa, J. Mendling, and H. A. Reijers, *Fundamentals of Business Process Management*. Springer, 2013.
- [16] K. Salimifard and M. Wright, "Petri net-based modelling of workflow systems: An overview," *European Journal of Operational Research*, vol. 134, no. 3, pp. 664–676, 2001.
- [17] Red Hat, "Using Ansible playbooks," 2023, last accessed 2023-03-01. [Online]. Available: https://docs.ansible.com/ansible/latest/playbook_guide/index.html
- [18] R. Stevens, D. Votipka, J. Dykstra, F. Tomlinson, E. Quartararo, C. Ahern, and M. L. Mazurek, "How Ready is Your Ready? Assessing the Usability of Incident Response Playbook Frameworks," in *CHI Conference on Human Factors in Computing Systems, 2022*, pp. 1–18.
- [19] J. Bollinger, B. Enright, and M. Valites, *Crafting the InfoSec playbook: security monitoring and incident response master plan*. O'Reilly Media, Inc., 2015.
- [20] I. Lella, E. Tsekmezoglou, R. S. Naydenov, C. Ciobanu, A. Malatras, and M. Theocharidou, "ENISA Threat Landscape 2022," European Union Agency for Network and Information Security (ENISA), Tech. Rep., 2022.
- [21] C. Lawson and A. Price, "2022 Market Guide for Security Orchestration, Automation and Response Solutions," Gartner, Tech. Rep., 2022.
- [22] MarketsandMarkets Research, "Security Orchestration Automation and Response (SOAR) Market Size, Share and Global Market Forecast to 2027," MarketsandMarkets, Tech. Rep., 2022.
- [23] OASIS, "CACAO Security Playbooks Version 1.0 - Committee Specification 02," OASIS, Tech. Rep., 2021, last accessed 2023-03-01. [Online]. Available: <https://docs.oasis-open.org/cacao/security-playbooks/v1.0/security-playbooks-v1.0.html>
- [24] FIRST Forum of Incident Response and Security Teams, "Automation SIG," 2023, last accessed 2023-03-01. [Online]. Available: <https://www.first.org/global/signs/automation/>
- [25] A. S. M. Rizvi, L. M. Bertholdo, J. M. Ceron, and J. S. Heidemann, "Anycast agility: Network playbooks to fight ddos," in *31st USENIX Security Symposium, USENIX Security 2022, Boston, MA, USA, August 10-12, 2022, 2022*, pp. 4201–4218.
- [26] M. A. Gurabi, A. Mandal, J. Popanda, R. Rapp, and S. Decker, "SASP: a Semantic web-based Approach for management of Shareable cybersecurity Playbooks," in *ARES 2022: The 17th International Conference on Availability, Reliability and Security, Vienna, Austria, August 23 - 26, 2022, 2022*, pp. 109:1–109:8.
- [27] D. Schlette, M. Caselli, and G. Pernul, "A Comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 4, pp. 2525–2556, 2021.
- [28] A. Shaked, Y. Cherdantseva, and P. Burnap, "Model-Based Incident Response Playbooks," in *ARES 2022: The 17th International Conference on Availability, Reliability and Security, Vienna, Austria, August 23 - 26, 2022, 2022*, pp. 26:1–26:7.
- [29] V. Mavroudis, P. Eis, M. Zádník, M. Caselli, and B. Jordan, "On the Integration of Course of Action Playbooks into Shareable Cyber Threat Intelligence," in *2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, December 15-18, 2021, 2021*, pp. 2104–2108.
- [30] C. Onwubiko and K. Ouazzane, "SOTER: A playbook for cybersecurity incident management," *IEEE Transactions on Engineering Management*, vol. 69, no. 6, pp. 3771–3791, 2020.
- [31] A. Applebaum, S. Johnson, M. Limiero, and M. Smith, "Playbook Oriented Cyber Response," in *2018 National Cyber Summit (NCS)*. IEEE, 2018, pp. 8–15.
- [32] X. Bouwman, V. L. Pochat, P. Foremski, T. van Goethem, C. H. Gañán, G. C. M. Moura, S. Tajalizadehkhooob, W. Joosen, and M. van Eeten, "Helping hands: Measuring the impact of a large threat intelligence sharing community," in *31st USENIX Security Symposium, USENIX Security 2022, Boston, MA, USA, August 10-12, 2022, 2022*, pp. 1149–1165.
- [33] B. Stojkovski, G. Lenzini, V. Koenig, and S. Rivas, "What's in a Cyber Threat Intelligence sharing platform?: A mixed-methods user experience investigation of MISP," in *ACSAC '21: Annual Computer Security Applications Conference, Virtual Event, USA, December 6 - 10, 2021, 2021*, pp. 385–398.

- [34] Financial Services Information Sharing and Analysis Center, "Reducing Cyber Risk Through Intelligence Sharing," 2023, last accessed 2023-03-01. [Online]. Available: <https://www.fsisac.com/who-we-are>
- [35] H. Paananen, M. Lapke, and M. Siponen, "State of the Art in Information Security Policy Development," *Computers & Security*, vol. 88, 2020.
- [36] K. J. Knapp, R. Franklin Morris, T. E. Marshall, and T. A. Byrd, "Information Security Policy: An Organizational-Level Process Model," *Computers & Security*, vol. 28, no. 7, pp. 493–508, 2009.
- [37] J. vom Brocke, S. Zelt, and T. Schmiedel, "On the role of context in business process management," *International Journal of Information Management*, vol. 36, no. 3, pp. 486–495, 2016.
- [38] M. Rosemann, J. Recker, and C. Flender, "Contextualisation of Business Processes," *International Journal of Business Process Integration and Management*, vol. 3, no. 1, pp. 47–60, 2008.
- [39] G. B. Willis and P. Royston, *Cognitive Interviewing: A Tool for Improving Questionnaire Design*. SAGE Publications, Inc., 2005.
- [40] V. Braun and V. Clarke, "Using thematic analysis in psychology," *Qualitative Research in Psychology*, vol. 3, no. 2, pp. 77–101, 2006.
- [41] D. S. Cruzes and T. Dybå, "Research synthesis in software engineering: A tertiary study," *Information and Software Technology*, vol. 53, no. 5, pp. 440–455, 2011.
- [42] V. Braun and V. Clarke, *Thematic Analysis: A Practical Guide*. London, UK: SAGE Publications, Ltd., 2019.
- [43] Object Management Group (OMG), "Business Process Model and Notation (BPMN) Specification Version 2.0.2," OMG, Tech. Rep., 2013, last accessed 2023-03-01. [Online]. Available: <https://www.omg.org/spec/BPMN/2.0.2/PDF>
- [44] D. Ashenden, "Information Security management: A human challenge?" *Information security technical report*, vol. 13, no. 4, pp. 195–201, 2008.
- [45] G. D. Bhatt, "Knowledge management in organizations: examining the interaction between technologies, techniques, and people," *Journal of knowledge management*, 2001.
- [46] B. Flyvbjerg, "Top ten behavioral biases in project management: An overview," *Project Management Journal*, vol. 52, no. 6, pp. 531–546, 2021.
- [47] T. D. Wilson and N. Brekke, "Mental contamination and mental correction: unwanted influences on judgments and evaluations," *Psychological bulletin*, vol. 116, no. 1, pp. 117–142, 1994.
- [48] D. Kahneman and A. Tversky, "Intuitive prediction: Biases and corrective procedures," Decisions and Designs Inc, Tech. Rep., 1977.
- [49] W. Tounsi and H. Rais, "A survey on technical threat intelligence in the age of sophisticated cyber attacks," *Computers & Security*, vol. 72, pp. 212–233, 2018.
- [50] F. Skopik, G. Settanni, and R. Fiedler, "A problem shared is a problem halved: A survey on the dimensions of collective cyber defense through security information sharing," *Computers & Security*, vol. 60, pp. 154–176, 2016.
- [51] S. Barnum, "Standardizing Cyber Threat Intelligence Information with the Structured Threat Information eXpression (STIX)," MITRE Corporation, Tech. Rep., 2012.
- [52] M. Wahnon, "Awesome Incident Response," 2023, last accessed 2023-03-01. [Online]. Available: <https://github.com/meirwah/awesome-incident-response>
- [53] Correlated Security, "Awesome SOAR," 2023, last accessed 2023-03-01. [Online]. Available: <https://github.com/correlatedsecurity/Awesome-SOAR>
- [54] Microsoft Community, "Windows security baselines," 2023, last accessed 2023-03-01. [Online]. Available: <https://learn.microsoft.com/en-us/windows/security/threat-protection/windows-security-configuration-framework/windows-security-baselines>
- [55] Center for Internet Security, "CIS Benchmarks," 2023, last accessed 2023-03-01. [Online]. Available: <https://www.cisecurity.org/cis-benchmarks>
- [56] US Department of Defense, "Security Technical Implementation Guides (STIGs)," 2023, last accessed 2023-03-01. [Online]. Available: <https://public.cyber.mil/stigs/>
- [57] F. Li and V. Paxson, "A large-scale empirical study of security patches," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 2201–2215.
- [58] OASIS, "Common Security Advisory Framework Version 2.0 OASIS Standard," OASIS, Tech. Rep., 2022, last accessed 2022-03-01. [Online]. Available: <https://docs.oasis-open.org/csaf/csaf/v2.0/csaf-v2.0.html>
- [59] A. Frisch, *Essential system administration: Tools and techniques for linux and unix administration*. O'Reilly Media, Inc., 2002.
- [60] E. Nemeth, G. Snyder, T. R. Hein, B. Whaley, and D. Mackin, *UNIX and Linux system administration handbook*. Pearson Education, 2018.
- [61] M. C. Langston, *Short Topics in Systems Administration - Documentation Writing for System Administrators*. USENIX/SAGE, 2003.
- [62] L. Hochstein and R. Moser, *Ansible: Up and Running: Automating configuration management and deployment the easy way*. O'Reilly Media, Inc., 2017.
- [63] K. Morris, *Infrastructure as code*. O'Reilly Media, Inc., 2020.
- [64] W. M. P. van der Aalst, "The Application of Petri Nets to Workflow Management," *Journal of Circuits, Systems, and Computers*, vol. 8, no. 1, pp. 21–66, 1998.
- [65] M. Weske, "Process Orchestrations," in *Business Process Management: Concepts, Languages, Architectures*. Springer, 2019, pp. 123–240.
- [66] Camunda, "The Process Orchestration Handbook," 2021, last accessed 2023-03-01. [Online]. Available: <https://camunda.com/process-orchestration/>
- [67] A. Oberweis, "An integrated approach for the specification of processes and related complex structured objects in business applications," *Decision Support Systems*, vol. 17, no. 1, pp. 31–53, 1996.
- [68] D. Georgakopoulos, M. F. Hornick, and A. P. Sheth, "An Overview of Workflow Management: From Process Modeling to Workflow Automation Infrastructure," *Distributed Parallel Databases*, vol. 3, no. 2, pp. 119–153, 1995.
- [69] P. O'Neill and A. S. Sohal, "Business Process Reengineering A review of recent literature," *Technovation*, vol. 19, no. 9, pp. 571–581, 1999.
- [70] OASIS, "Open Command and Control (OpenC2) Language Specification Version 1.0 - Committee Specification 02," OASIS, Tech. Rep., 2019, last accessed 2023-03-01. [Online]. Available: <https://docs.oasis-open.org/openc2/oc2ls/v1.0/oc2ls-v1.0.html>

Appendix A. Playbook terminology

TABLE 7. SECURITY STANDARDS AND PROCESS NOTATIONS HELP TO UNDERSTAND PLAYBOOK TERMINOLOGY.

Term	Description	CACAO [23]	OpenC2 [70]	CSAF [58]	BPMN [43]
Playbook	A playbook describes a specific cybersecurity process or procedure based on a workflow with individual steps or actions. Playbooks also include metadata. (Example: Phishing playbook)	Playbook: “[...] a playbook consisting of one or more security actions combined into a sequence or algorithmically-defined use.” “A template playbook will not be immediately executable by a receiving organization but may inform their own executable playbook for their specific environment or organization.”	N/A	CSAF document: “security advisory text document [...].” Advisory: “reporting item that describes a condition present in an artifact and that requires action by the consumers.”	BPMN Model / BPMN Diagram: “[...] a BPMN diagram is a particular snapshot of a BPMN model at a certain point in time.”
Workflow and Workflow Step	A workflow or course of action captures multiple workflow steps and procedural logic. A workflow step is defined by its position and its structural components. (Example: Start - 1) Step X, 2) Step Y, 3) Step Z - End)	Workflows contain a series of steps [...]. Workflows process steps either sequentially, in parallel, or both depending on the type of steps required by the playbook.”	OpenC2 Command: “The Command describes an Action to be performed on a Target and may include information identifying the Actuator or Actuators that are to execute the Command.” Command \approx workflow step	Remediations: “Every Remediation item [...] specifies details on how to handle (and presumably, fix) a vulnerability.” Remediations - Details: “[...] contains a thorough human-readable discussion of the remediation.”	Business Process: “A defined set of business activities that represent the steps required to achieve a business objective. It includes the flow and use of information and resources.” Activity: “Work that a company or organization performs using business processes. An activity can be atomic or non-atomic (compound).”
Actuator	An actuator represents an entity performing an action. Information systems, applications, or humans are actuators and provide specific capabilities. (Example: Incident handler / linux server)	Targets: “The CACAO target object contains detailed information about the entities or devices that accept, receive, process, or execute one or more commands as defined in a workflow step. Targets contain the information needed to send commands as defined in steps to devices or humans” (e.g., individual, ssh, http-api, net-address).	Actuator: “The Actuator executes the Command. The Actuator will be defined within the context of an Actuator Profile” (e.g., Stateless Paket Filtering Profiler-function).	N/A	Pool: “A Pool represents a Participant in a Collaboration.” Lane: “A partition that is used to organize and categorize activities within a Pool. [...] Lanes are often used for such things as internal roles (e.g., Manager, Associate), systems (e.g., an enterprise application), or an internal department (e.g., shipping, finance).”
Action	An action is an executable instruction or task representing a precise cybersecurity measure. Actions are manifold but center on operations towards systems, networks, or humans. (Example: Investigate / block)	Action: “[...] security activity in an organization [...]. Those actions may represent an activity to investigate, prevent, mitigate or remediate a specific security state that has either occurred or the organization is taking action to ensure the security state never occurs.” Commands: “The CACAO command object [...] contains detailed information about the commands that are to be executed or processed automatically or manually as part of a workflow step” (e.g., last; netstat -n; ls -l -a /root or Disconnect the machine from the network and call the SOC on-call person).	Action: “The task or activity to be performed” (e.g., scan, deny).	N/A	Task: “An atomic activity that is included within a Process. A Task is used when the work in the Process is not broken down to a finer level of Process Model detail. Generally, an end-user, an application, or both will perform the Task.”
Artifact	An artifact serves as the object or input of an action. Artifacts allow identification and refer to threat intelligence. (Example: IP address)	Variables can be defined and used as the playbook is executed” (e.g., \$\$ip4-addr, \$\$sha256-hash). in_args: “The optional list of arguments passed to the target(s) as input to the step.” Variables and in_args \approx artifact	Target: “The object of the action. The Action is performed on the Target” (e.g., ip4_net, device, file, mac_addr).	N/A (related to Vulnerabilities Property - Remediations - Group Ids and Vulnerabilities Property - Remediations - Product Ids “[...] the current remediation item applies to.”)	Data Objects: “The primary construct for modeling data within the Process flow is the DataObject element. A DataObject has a well-defined lifecycle, with resulting access constraints.” Data Inputs: “Data requirements are captured as Data Inputs [...].”
Output	An output is the result of a workflow step. Outputs are optional and can contain status codes or other data. (Example: Success)	out_args: “The optional list of arguments that are returned from this step after execution of the commands by the targets.”	OpenC2 Response: “The Response is a Message sent from the recipient of a Command. Response messages provide acknowledgment, status, results from a query, or other information.”	N/A	Data Objects: “The primary construct for modeling data within the Process flow is the DataObject element. A DataObject has a well-defined lifecycle, with resulting access constraints.” Data Outputs: “Data that is produced is captured using Data Outputs [...].”

Appendix B. Graphical phishing playbook

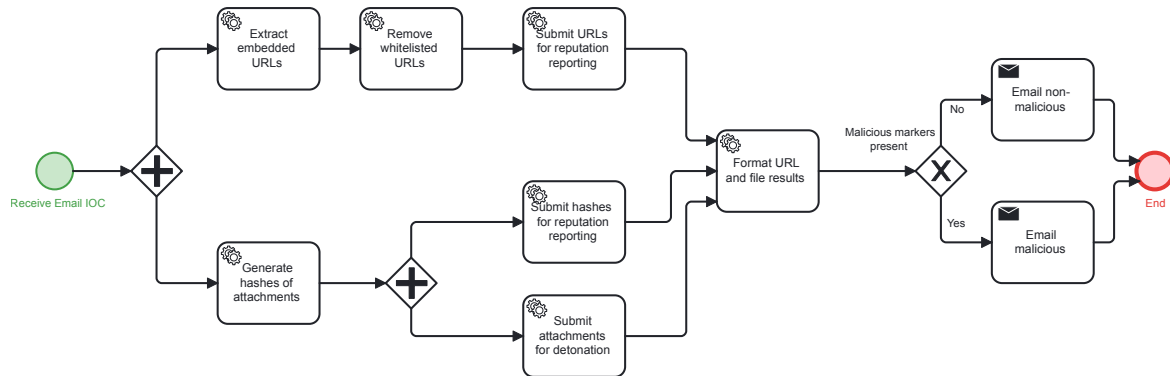


Figure 5. Graphical visualization of a phishing playbook with BPMN (source: IACD).

Appendix C. Code-based phishing playbook workflow step (excerpt)

```

{
  "action": "URL reputation",
  "action_type": "investigate",
  "assets": [
    {
      "app_name": "VirusTotal",
      "app_version": "1.2.40",
      "output": [
        {
          "data_path": "action_result.status"
        },
        {
          "data_path": "action_result.parameter.url"
        },
        {
          "data_path": "action_result.data.*.resource"
        },
        {
          "data_path": "action_result.data.*.scan_date"
        },
        {
          "data_path": "action_result.data.*.scan_id"
        },
        {
          "data_path": "action_result.data.*.url"
        },
        {
          "data_path": "action_result.message"
        }
      ],
      "parameters": {
        "url": {
          "data_type": "string",
          "description": "URL to query",
          "key": "url",
          "required": true
        }
      }
    }
  ]
}

```

Figure 6. Excerpt of a workflow step investigating URL reputation within a code-based phishing playbook (source: Splunk SOAR).

Appendix D. Interview codebook

The codebook matches interview questions to external and internal influencing factors. Most factors relate to the three incident scenarios (i.e., Ransomware, DDoS, APT) and are addressed by what-if questions. In addition, we used the overarching questions for each scenario. The overarching questions provided in-depth results emphasizing the relevance of individual factors.

Overarching Questions:

- How does your organization handle a [incident type] scenario?
- How does your organization handle this scenario differently than other organizations?

External Factor 1: Attacker characteristics (motivation, behavior, location)

What would your organization do differently if ...

- it is operating in Israel/Ukraine?
- it assumes a state-sponsored attacker group located in Iran/Russia behind the attack?
- it has specified incident response time limits for APT attacks to 1 month?

External Factor 2: Industry standards and guidelines (including frameworks)

Does your organization use ...

- generic security process descriptions (e.g., BPMN, NIST Incident Response Life Cycle, NIST Cybersecurity Framework, FIRST CSIRT Framework, etc.)?
- incident response maturity models (e.g., SIM3, etc.)?
- incident response standards (e.g., CACAO, OpenC2, RE&CT, MITRE D3FEND, etc.)?
- an incident response policy to define guiding principles?

External Factor 3: Laws and regulations (business structure, location/privacy, sector)

What would your organization do differently if ...

- it is a publicly traded company in the United States?
- it has customers in the European Union?
- it is a sub-contractor for the defense industry?

External Factor 4: Supply chain and business partners expectations

What would your organization do differently if ...

- it has sourced server hosting to a third party?
- it is a sub-contractor for the defense industry?
- it observed compromised email accounts?

Internal Factor 1: Incident response directives (data operations, attack targets/assets)

What would your organization do differently if ...

- it has proxy log retention set to 1 week?
- it has business-critical applications running on the targeted web server?
- it has defined formal reporting requirements?
- its CFO's laptop is affected?

Internal Factor 2: People (security culture/mandate, security team)

What would your organization do differently if ...

- it has a dedicated Cyber Threat Intelligence team?
- it has defined email for incident response communication?
- it has a security team with 100 security experts?
- it has specified to contact the CISO, but the CISO is unavailable?

Internal Factor 3: Technology (infrastructure/tech stack, security tools)

What would your organization do differently if ...

- it defined an incident budget of \$100k?
- it has sourced server hosting to a third party?
- it has a Web Application Firewall and additional server capacity?

Does your organization use ...

- Security Orchestration, Automation and Response (SOAR) tools (e.g., Splunk Phantom, Cortex XSOAR, Tines, etc.)?

Curriculum Vitae

DANIEL SCHLETTE

Chair of Information Systems (IFS)
Faculty of Business, Economics, Management Information Systems
University of Regensburg, Germany

EDUCATION

2019 – 2023	PhD Student <i>University of Regensburg, Germany</i>
2016 – 2019	M.Sc. Management Information Systems (Honors) <i>University of Regensburg, Germany</i>
2017	Exchange Program — Information Systems <i>Kingston University London, UK</i>
2013 – 2016	B.Sc. Management Information Systems (Honors Program) <i>University of Regensburg, Germany</i>

RESEARCH PROJECTS

Siemens. Responsible for the research project *Contributions to Cyber Defence* fostering the integration of data formats in Cyber Threat Intelligence (2019-2022).

DEVISE. Responsible for a successful project proposal and research on assessing threat intelligence quality within a 3-year research project funded by BMBF (2021-2023).

TEACHING

2019 – 2023	Co-Lecturer – Security of data-intensive Applications <i>Graduate lecture at University of Regensburg</i>
2021 – 2022	Tutor – Information Systems - Developments and Trends <i>Graduate lecture at University of Regensburg</i>
2021 – 2022	Tutor – Corporate Databases <i>Undergraduate lecture at University of Regensburg</i>
2016 – 2018	Student Tutor – Corporate Databases <i>Undergraduate lecture at University of Regensburg</i>

REVIEWING ACTIVITIES

Conferences. DBSec 2023, ARES 2023, WI 2023, ISPEC 2022, ESORICS 2022/2, ICICS 2022, ARES 2022, COMPSAC 2022, ESORICS 2022/1, WISE 2022, CAiSE 2022, ESORICS 2021, TrustBus 2021, DBSec 2021, SECRIPT 2021, ARES 2021, NSS 2020, TrustCom 2020, SPBP 2020, ESORICS 2020, ARES 2020, TrustBus 2020, CPSS 2020, DBSec 2020, CAiSE 2020, ISPEC 2019

Journals. International Journal of Information Security (2x), Computers & Security

PUBLICATIONS

- [1] SCHLETTE, D., BÖHM, F., CASELLI, M., & PERNUL, G. (2021). Measuring and visualizing cyber threat intelligence quality. *International Journal of Information Security*, 20(1), pp. 21-38.
- [2] SCHLETTE, D., MENGES, F., BAUMER, T., & PERNUL, G. (2020). Security Enumerations for Cyber-Physical Systems. In *Data and Applications Security and Privacy XXXIV: 34th Annual IFIP WG 11.3 Conference (DBSec 2020)*, *Lecture Notes in Computer Science 12122*, pp. 64-76.
- [3] SCHLETTE, D. (2021). Cyber Threat Intelligence. In *Encyclopedia of Cryptography, Security and Privacy (3rd Edition)*, pp. 1-3.
- [4] SCHLETTE, D. (2021). Cyber Threat Intelligence Sharing. In *Encyclopedia of Cryptography, Security and Privacy (3rd Edition)*, pp. 1-3.
- [5] SCHLETTE, D., VIELBERTH, M., & PERNUL, G. (2021). CTI-SOC2M2—The quest for mature, intelligence-driven security operations and incident response capabilities. *Computers & Security*, 111, 102482, pp. 1-20.
- [6] SCHLETTE, D., CASELLI, M., & PERNUL, G. (2021). A Comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective. *IEEE Communications Surveys & Tutorials*, 23(4), pp. 2525-2556.
- [7] EMPL, P., SCHLETTE, D., ZUPFER, D., & PERNUL, G. (2022). SOAR4IoT: Securing IoT Assets with Digital Twins. In *The 17th International Conference on Availability, Reliability and Security (ARES 2022)*, pp. 4:1-4:10.
- [8] DIETZ, M., SCHLETTE, D., & PERNUL, G. (2022). Harnessing Digital Twin Security Simulations for systematic Cyber Threat Intelligence. In *46th Annual Computers, Software, and Applications Conference (COMPSAC 2022)*, pp. 789-797.
- [9] EMPL, P., SCHLETTE, D., STÖGER, L., & PERNUL, G. (2023). Generating ICS Vulnerability Playbooks with Open Standards. Submitted to *The 18th International Conference on Availability, Reliability and Security (ARES 2023)*.
- [10] SCHLETTE, D., EMPL, P., CASELLI, M., SCHRECK, T., & PERNUL, G. (2024). Do you Play It by the Books? A Study on Incident Response Playbooks and Influencing Factors. Submitted to *The 45th IEEE Symposium on Security and Privacy (S&P 2024)*.

TALKS

2022	SOAR4IoT: Securing IoT Assets with Digital Twins <i>ARES Conference 2022</i>
2022	Beyond Incident Reporting – An Analysis of Structured Representations for Incident Response <i>Annual FIRST Conference 2022</i>
2020	Security Enumerations for Cyber-Physical Systems <i>DBSec Conference 2020 (virtual)</i>