This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2021.3117338, IEEE Communications Surveys & Tutorials

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

Daniel Schlette, Marco Caselli and Günther Pernul

***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 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.

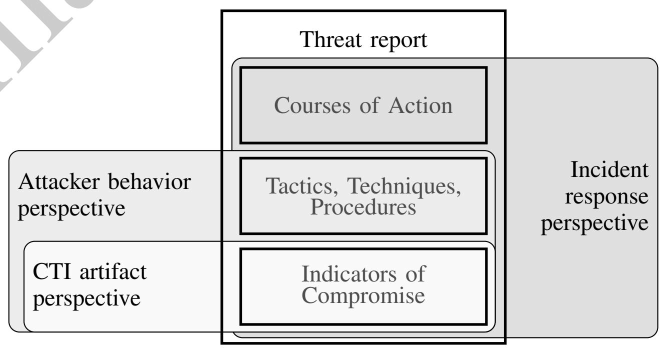

Fig. 1. Cyber Threat Intelligence Perspectives

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 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 renown frameworks – the incident

D. Schlette and G. Pernul are with the Chair of Information Systems, University of Regensburg, Universitätsstr. 31, 93053 Regensburg, Germany (e-mail: daniel.schlette@ur.de); M. Caselli is with Siemens AG, Otto-Hahn-Ring 6, 81739 Munich, Germany

Manuscript received February 1, 2021; accepted September 29, 2021

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

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 [27], [21]. 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.

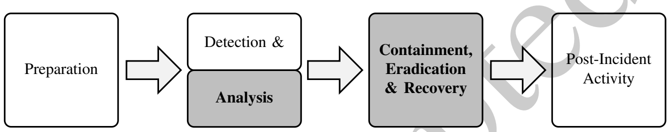

Fig. 2. Survey focus based on NIST Incident Response Life Cycle [23]

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 US *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*, *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

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2021.3117338, IEEE Communications Surveys & Tutorials

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 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.

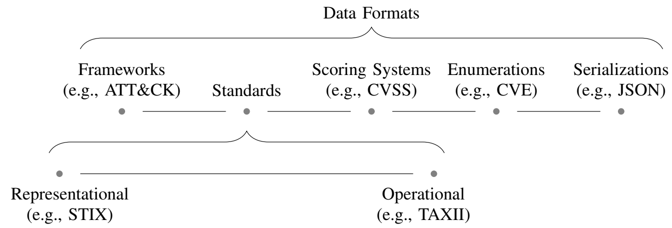

Fig. 3. Categorization of Cyber Threat Intelligence Data Formats

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*

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

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

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 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.

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2021.3117338, IEEE Communications Surveys & Tutorials

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	DFLabs	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	×

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).

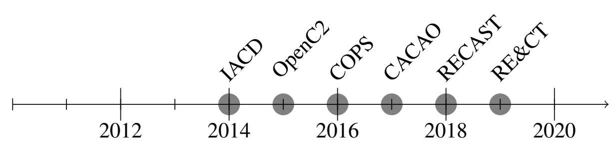

Fig. 4. Timeline of Incident Response Formats (first mention)

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 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

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2021.3117338, IEEE Communications Surveys & Tutorials

different comparison criteria [7], [6], [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.

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
	Unambiguous Semantics	Clarity, Interorganizational Understanding and Application
Structural	Workflow	Sequencing, Operations
	Actuator	Actionability
	Action	Atomicity
	Artifact	CTI Integration
Technological	Community	Usability, Acceptance, Maintenance
	Application	Technical Integration, Interoperability
	Serialization	Data Storage, Data Transfer
Security	Confidentiality	Information Sharing, Operations
	Authorization	Misuse Prevention, Operations
	Prioritization	Information Importance, Operations

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 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).

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

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.

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

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 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.

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2021.3117338, IEEE Communications Surveys & Tutorials

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.

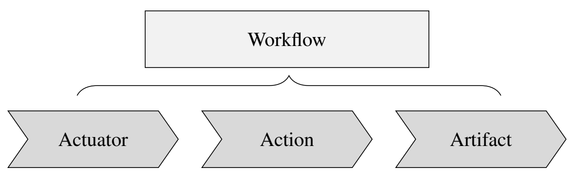

Fig. 5. Structural Incident Response Concepts

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

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

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 oftentimes 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 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.

TABLE V
Incident Response Format Analysis Approach

Category	Description	Level
Name	Descriptive term	Basics
Abbreviation	Descriptive, short identifier	
Main objective	Distilled overall objective	
Inception	Year of first publication	
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	

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

[1]https://www.json.org
[2]https://yaml.org/spec/1.2/spec.html

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

the nonprofit Organization for the Advancement 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.

Negative effects when using the CACAO format caused by ambiguity are reduced with a terminology section and

[3]https://datatracker.ietf.org/wg/cacao/about/

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

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

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.

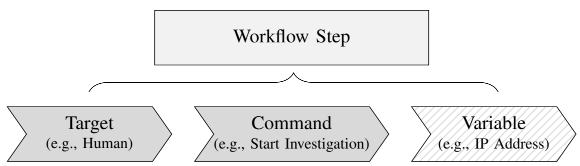

Fig. 6. Structural Description of CACAO Playbooks

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 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 COPS[5] and Demisto content[6] 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 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 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

[5]https://github.com/demisto/COPS
[6]https://github.com/demisto/content

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2021.3117338, IEEE Communications Surveys & Tutorials

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 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.

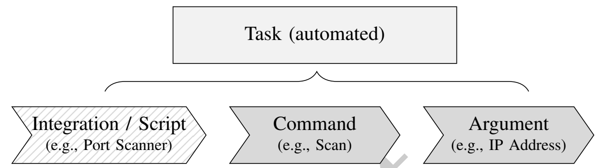

Fig. 7. Structural Description of COPS Playbooks

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 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.

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

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 website[7] 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, [100] and [42] 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. 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

[7] https://www.iacdautomate.org/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2021.3117338, IEEE Communications Surveys & Tutorials

`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 exists 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 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.

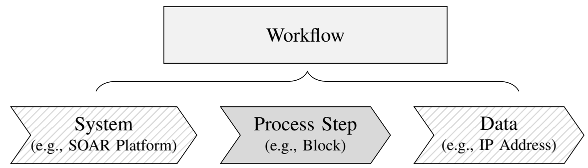

Fig. 8. Structural Description of IACD Playbooks / Workflows

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 US 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.

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

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. [120], [100], [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 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

[8] https://openc2.org/

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. 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.

Unambigous 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.

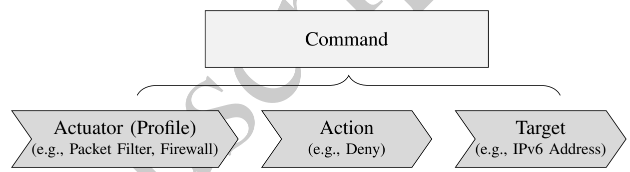

Fig. 9. Structural Description of OpenC2 Messages

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 × targets) [117]. Within actuator profiles, specifiers are defined 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 con-

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

clude 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.

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2021.3117338, IEEE Communications Surveys & Tutorials

Statistics: Information about the RE&CT incident response format can be retrieved from the RE&CT[9] 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 $\times$ 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.

Umambiguous 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.

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

[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/

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2021.3117338, IEEE Communications Surveys & Tutorials

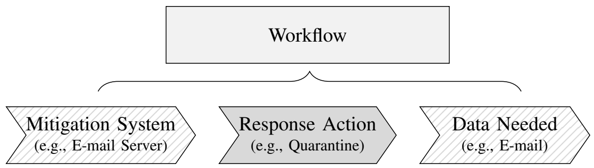

Fig. 10. Structural Description of RE&CT Playbooks

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

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

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 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.

Umambiguous 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.

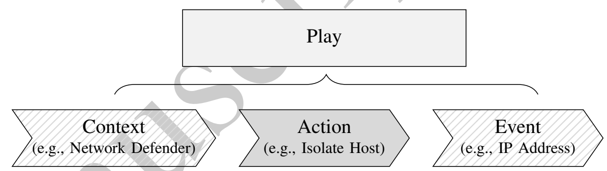

Fig. 11. Structural Description of RECAST Playbooks

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 *Actuator* 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.

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

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 [126], [67]. Analogous, Digital Forensic eXtensible Markup Language (DFXML) has the objective to describe digital forensic information and the results of digital forensic processing [127], [68]. 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 [129], [35]. 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 [84], [90], [96], [80], [79] 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 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 A 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

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

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.

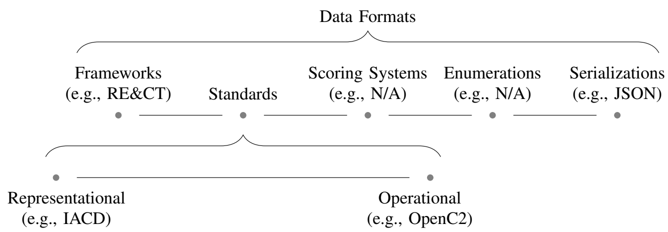

Fig. 12. Categorization of Incident Response Automation Data Formats

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, 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

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

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 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.

TABLE VII
COMPARISON OF STRUCTURAL CONCEPT IMPLEMENTATION

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

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2021.3117338, IEEE Communications Surveys & Tutorials

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.

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	×

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.

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

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

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

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.

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

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.

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2021.3117338, IEEE Communications Surveys & Tutorials

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 (cf. 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 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.

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

APPENDIX A
INCIDENT RESPONSE FORMAT ANALYSIS

APPENDIX B
ACRONYMS

ACKNOWLEDGMENT

Part of this research was supported by the Federal Ministry of Education and Research, Germany, as part of the BMBF DEVISE project. Part of this project has received funding from the European Union's Horizon 2020 research and innovation program under Grant Agreement No. 830927.

REFERENCES

[1] R. McMillan, "Definition: Threat intelligence," 2013, last accessed 2020-10-20. [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, CERT-UK, Tech. Rep., 2015.
[3] R. A. Martin, "Making security measurable and manageable," in *MILCOM 2008-2008 IEEE Military Communications Conference*. IEEE, 2008, pp. 1–9.
[4] 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.
[5] L. Dandurand, A. Kaplan, P. Kácha, Y. Kadobayashi, A. Kompanek, T. Lima, T. Millar, J. Nazario, R. Perlotto, and W. Young, "Standards and tools for exchange and processing of actionable information," European Union Agency for Network and Information Security (ENISA), Tech. Rep., 2014.
[6] J. Steinberger, A. Sperotto, M. Golling, and H. Baier, "How to exchange security events? overview and evaluation of formats and protocols," in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. IEEE, 2015, pp. 261–269.
[7] F. Menges and G. Pernul, "A comparative analysis of incident reporting formats," *Computers & Security*, vol. 73, pp. 87–101, 2018.
[8] C. Wagner, A. Dulaunoy, G. Wagener, and A. Iklody, "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'16*, S. Katzenbeisser, E. Weippl, E.-O. Blass, and F. Kerschbaum, Eds. ACM, 2016, pp. 49–56.
[9] C. Sauerwein, C. Sillaber, A. Mussmann, and R. Breu, "Threat intelligence sharing platforms: An exploratory study of software vendors and research perspectives," in *Proceedings of the 13th International Conference on Wirtschaftsinformatik (WI 2017)*, J. M. Leimeister and W. Brenner, Eds., 2017, pp. 837–851.
[10] F. Menges, B. Putz, and G. Pernul, "DEALER: decentralized incentives for threat intelligence reporting and exchange," *International Journal of Information 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," *Computers & Security*, vol. 60, pp. 154–176, 2016.
[12] V. Mavroeidis and S. Bromander, "Cyber threat intelligence model: An

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

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/COMST.2021.3117338, IEEE Communications Surveys & Tutorials

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

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

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

evaluation of taxonomies, sharing standards, and ontologies within cti," in *2017 European Intelligence and Security Informatics Conference (EISIC)*. IEEE, 2017, pp. 91–98.

[13] C. Sillaber, C. Sauerwein, A. Mussmann, and R. Breu, "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*, 2016, pp. 65–70.

[14] D. Schlette, F. Böhm, M. Caselli, and G. Pernul, "Measuring and visualizing cyber threat intelligence quality," *International Journal of Information Security*, pp. 1–18, 2020.

[15] V. G. Li, M. Dunn, P. Pearce, D. McCoy, G. M. Voelker, and S. Savage, "Reading the tea leaves: A comparative analysis of threat intelligence," in *28th {USENIX} Security Symposium ({USENIX} Security 19)*, 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 Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 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," *Computers & Security*, vol. 95, p. 101867, 2020.

[18] A. Berndt and J. Ophoff, "Exploring the value of a cyber threat intelligence function in an organization," in *IFIP World Conference on Information Security Education*. Springer, 2020, pp. 96–109.

[19] D. Preuveneers, W. Joosen, J. Bernal Bernabe, and A. Skarmeta, "Distributed security framework for reliable threat intelligence sharing," *Security and Communication Networks*, vol. 2020, 2020.

[20] J. D. Howard and T. A. Longstaff, "A common language for computer security incidents," Sandia National Labs, Tech. Rep., 1998.

[21] N. H. Ab Rahman and K.-K. R. Choo, "A survey of information security incident handling in the cloud," *Computers & Security*, vol. 49, pp. 45–69, 2015.

[22] C. Islam, M. A. Babar, and S. Nepal, "A multi-vocal review of security orchestration," *ACM Computing Surveys (CSUR)*, vol. 52, no. 2, pp. 1–45, 2019.

[23] P. Cichonski, T. Millar, T. Grance, and K. Scarfone, "Computer security incident handling guide," *NIST Special Publication*, vol. 800, no. 61, pp. 1–147, 2012.

[24] C. Alberts, A. Dorofee, G. Killcrece, R. Ruefle, and M. Zajicek, "Defining incident management processes for CSIRTs: A work in progress," Carnegie Mellon University Software Engineering Institute, Tech. Rep., 2004.

[25] J. Van Bon, A. De Jong, A. Kolthof, M. Pieper, R. Tjassing, A. van der Veen, and T. Verheijen, *Foundations of IT Service Management Based on ITIL®*. Van Haren, 2008, vol. 3.

[26] F. C. Freiling and B. Schwittay, "A common process model for incident response and computer forensics," *IMF 2007: IT-Incident Management & IT-Forensics*, 2007.

[27] B. Grobauer and T. Schreck, "Towards incident handling in the cloud: challenges and approaches," in *Proceedings of the 2010 ACM workshop on Cloud computing security workshop*, 2010, pp. 77–86.

[28] G. D. Bhatt, "Knowledge management in organizations: examining the interaction between technologies, techniques, and people," *Journal of knowledge management*, 2001.

[29] B. Schneier, "The future of incident response," *IEEE Security & Privacy*, vol. 12, no. 5, pp. 96–96, 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. 227 756–227 779, 2020.

[31] M. J. West-Brown, D. Stikvoort, K.-P. Kossakowski, G. Killcrece, and R. Ruefle, "Handbook for computer security incident response teams (csirts)," Carnegie Mellon University Software Engineering Institute, Tech. Rep., 2003.

[32] C. Zimmerman, "Cybersecurity operations center," *The MITRE Corporation*, 2014.

[33] Executive Office of the President, "Executive order 14028 of may 12, 2021 – improving the nation's cybersecurity," 2021, last accessed 2021-09-01. [Online]. Available: https://www.federalregister.gov/d/2021-10460/p-113

[34] M. Bromiley, "Empowering incident response via automation," *SANS Institute InfoSec Reading Room*, 2019.

[35] C. Neiva, C. Lawson, T. Bussa, and G. Sadowski, "2020 market guide for security orchestration, automation and response solutions," Gartner, Tech. 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 Communications Surveys & Tutorials*, vol. 20, no. 2, pp. 1361–1396, 2017.

[37] OASIS, *CACAO Security Playbooks Version 1.0 - Committee Specification 01*, B. Jordan and A. Thomson, Eds. OASIS, 2021, last accessed 2021-01-15. [Online]. Available: https://docs.oasis-open.org/cacao/security-playbooks/v1.0/security-playbooks-v1.0.html

[38] DEMISTO, "COPS - Collaborative Open Playbook Standard," 2018, last accessed 2020-12-15. [Online]. Available: https://github.com/demisto/COPS

[39] IACD, "Integrated Adaptive Cyber Defense (IACD) Playbooks – a specification for defining, building and employing playbooks to enable cybersecurity integration and automation," 2017, last accessed 2020-10-15. [Online]. Available: https://www.iacdautomate.org/s/IACD-Playbook-Thin-Specification.pdf

[40] OASIS, *Open Command and Control (OpenC2) Language Specification Version 1.0 - Committee Specification 02*, J. Romano and D. Sparrell, Eds. OASIS, 2020, last accessed 2020-11-15. [Online]. Available: https://docs.oasis-open.org/openc2/oc2ls/v1.0/cs02/oc2ls-v1.0-cs02.html

[41] ATC Project, "RE&CT framework documentation," 2020, last accessed 2020-10-01. [Online]. Available: https://atc-project.github.io/atc-react/

[42] A. Applebaum, S. Johnson, M. Limiero, and M. Smith, "Playbook oriented cyber response," in *2018 National Cyber Summit (NCS)*. IEEE, 2018, pp. 8–15.

[43] M. Liu, Z. Xue, X. He, and J. Chen, "Cyberthreat-intelligence information sharing: Enhancing collaborative security," *IEEE Consumer Electronics Magazine*, vol. 8, no. 3, pp. 17–22, 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," *Leading Issues in Information Warfare & Security Research*, 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," The MITRE Corporation, Tech. Rep., 2018.

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

[46] A. Dulaunoy and A. Iklody, "MISP core format," Internet Engineering Task Force (IETF), Tech. Rep., 2020, last accessed 2020-10-20. [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, Tech. Rep., 2014. [Online]. Available: http://stixproject.github.io/getting-started/whitepaper/

[48] OASIS Cyber Threat Intelligence (CTI) Technical Committee, *STIX™ Version 2.1: Committee Specification 01*, B. Jordan, R. Piazza, and T. Darley, Eds. OASIS, 2020, last accessed 2020-10-20. [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, Tech. Rep., 2014. [Online]. Available: https://taxiiproject.github.io/getting-started/whitepaper/

[50] OASIS Cyber Threat Intelligence (CTI) Technical Committee, *TAXII™ Version 2.1: Committee Specification 01*, B. Jordan and D. Varner, Eds. OASIS, 2020, last accessed 2020-10-20. [Online]. Available: https://docs.oasis-open.org/cti/taxii/v2.1/taxii-v2.1.html

[51] Forum of Incident Response and Security Teams (FIRST), "Common Vulnerability Scoring System version 3.1: Specification document - revision 1," 2019, last accessed 2020-11-01. [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," National Institute of Standards and Technology, Maryland, USA, Tech. Rep., 2011, NIST Interagency Report 7695.

[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 *Recent Advances in Intrusion Detection*, vol. 7, 1999, p. 9.

[54] MITRE, "Common Weakness Enumeration - a community-developed list of software & hardware weakness types," 2020, last accessed 2020-11-01. [Online]. Available: https://cwe.mitre.org/index.html

[55] S. Fenz, A. Ekelhart, and E. Weippl, "Semantic potential of existing security advisory standards," in *Proceedings of the FIRST 2008 Conference-Forum of Incident Response and Security Teams*, 2008.

[56] J. L. Hernandez-Ardieta, J. E. Tapiador, and G. Suarez-Tangil, "Information sharing models for cooperative cyber defence," in *2013 5th International Conference on Cyber Conflict (CYCON 2013)*. IEEE, 2013, pp. 1–28.

[57] L. Dandurand, A. Kaplan, P. Kácha, Y. Kadobayashi, A. Kompanek, T. Lima, T. Millar, J. Nazario, R. Perlotto, and W. Young, "Standards and tools for exchange and processing of actionable information," 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. Costa Gondim, R. de Oliveira Albuquerque, and L. J. García 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. Breu, "Towards an evaluation framework for threat intelligence sharing platforms," in *Proceedings of the 53rd Hawaii International Conference on System Sciences*, 2020.

[62] T. D. Wagner, K. Mahbub, E. Palomar, and A. E. Abdallah, "Cyber threat intelligence sharing: Survey and research directions," *Computers & Security*, vol. 87, p. 101589, 2019.

[63] Red Hat, "Ansible - Ansible is simple IT automation," 2020, last accessed 2020-10-20. [Online]. Available: https://www.ansible.com/

[64] Object Management Group (OMG), "Business Process Model and Notation (BPMN) specification version 2.0," 2011, last accessed 2020-10-20. [Online]. Available: https://www.omg.org/spec/BPMN/2.0/PDF

[65] McAfee, "OpenDXL - open data exchange layer," 2016, last accessed 2020-10-20. [Online]. Available: https://www.opendxl.com/

[66] J. Field, S. Banghart, and D. Waltermire, "RFC 8322 - resource-oriented lightweight information exchange (ROLIE)," Internet Engineering Task Force (IETF), Tech. Rep., 2018, last accessed 2020-10-20. [Online]. Available: https://tools.ietf.org/html/rfc8322

[67] M. Cohen and B. Schatz, "AFF4 standard v1.0," 2017, last accessed 2020-10-20. [Online]. Available: https://github.com/aff4/Standard

[68] DFXML Working Group, "DFXML schema version 1.2.0," 2017. [Online]. Available: https://github.com/dfxml-working-group/dfxml_schema

[69] Micro Focus, "ArcSight SOAR," 2020, last accessed 2020-10-20. [Online]. Available: https://www.microfocus.com/en-us/products/arcsight-soar/overview

[70] Ayehu, "Ayehu - next-gen IT automation platform powered by AI," 2020, last accessed 2020-10-20. [Online]. Available: https://ayehu.com/

[71] Palo Alto Networks, "Cortex XSOAR - security orchestration, automation and response (SOAR)," 2020, last accessed 2020-10-20. [Online]. Available: https://www.paloaltonetworks.com/cortex/xsoar

[72] D3 Security, "D3 SOAR - security orchestration and automated incident response with MITRE ATT&CK," 2020, last accessed 2020-10-20. [Online]. Available: https://d3security.com/

[73] Dragos, "The Dragos platform," 2016, last accessed 2020-10-20. [Online]. Available: https://www.dragos.com/platform/

[74] EclecticIQ, "EclecticIQ - threat intelligence powered cybersecurity," 2020, last accessed 2020-10-20. [Online]. Available: https://www.eclecticiq.com/

[75] Fortinet, "FortiSOAR - security orchestration, automation and response (SOAR)," 2020, last accessed 2020-10-20. [Online]. Available: https://www.fortinet.com/products/fortisoar

[76] FireEye, "Helix security platform," 2020, last accessed 2020-10-20. [Online]. Available: https://www.fireeye.com/products/helix.html

[77] DFLabs, "IncMan SOAR - automate," 2020, last accessed 2020-10-20. [Online]. Available: https://www.dflabs.com/

[78] Rapid7, "Security orchestration and automation with InsightConnect," 2020, last accessed 2020-10-20. [Online]. Available: https://www.rapid7.com/products/insightconnect/

[79] The Linux Foundation, "ONAP - Open Network Automation Platform," 2020, last accessed 2020-10-20. [Online]. Available: https://www.onap.org/

[80] Palo Alto Networks - Unit 42, "Unit 42 Playbook Viewer," 2020, last accessed 2020-11-01. [Online]. Available: https://pan-unit42.github.io/playbook_viewer/

[81] IBM Security, "IBM Resilient security orchestration, automation and response (SOAR)," 2020, last accessed 2020-10-20. [Online]. Available: https://www.ibm.com/products/resilient-soar-platform

[82] Resolve, "Resolve - accelerate security incident response with automation and orchestration," 2020, last accessed 2020-10-20. [Online]. Available: https://resolve.io/it-automation-resources/accelerate-security-incident-response-with-automation-soar

[83] ServiceNow, "SecOps - enterprise security operations," 2020, last accessed 2020-10-20. [Online]. Available: https://www.servicenow.com/products/security-operations.html

[84] F. Ødegårdstuen, "Shuffle SOAR," 2020, last accessed 2020-10-20. [Online]. Available: https://shuffler.io/

[85] Siemplify, "Siemplify - security orchestration, automation & response (SOAR) platform," 2020, last accessed 2020-10-20. [Online]. Available: https://www.siemplify.co/

[86] LogicHub, "The SOAR+ platform," 2020, last accessed 2020-10-20. [Online]. Available: https://www.logichub.com/

[87] Honeycomb, "Honeycomb SOCAutomation," 2020, last accessed 2020-10-20. [Online]. Available: https://socautomation.com/

[88] Splunk, "Splunk Phantom security orchestration & automation," 2020, last accessed 2020-10-20. [Online]. Available: https://www.splunk.com/en_us/software/splunk-security-orchestration-and-automation.html

[89] Swimlane, "Swimlane - security orchestration, automation and response platform," 2020, last accessed 2020-10-20. [Online]. Available: https://swimlane.com/

[90] TheHive Project, "TheHive & Cortex - a 4-in-1 security incident response platform," 2020, last accessed 2020-10-20. [Online]. Available: https://thehive-project.org/

[91] ThreatConnect, "ThreatConnect - security orchestration, automation, and response platform," 2020, last accessed 2020-10-20. [Online]. Available: https://threatconnect.com/solution/security-orchestration-automation-response/

[92] Anomali, "ThreatStream - threat intelligence platform," 2020, last accessed 2020-10-20. [Online]. Available: https://www.anomali.com/products/threatstream

[93] ThreatQuotient, "ThreatQ - threat intelligence platform," 2020, last accessed 2020-10-20. [Online]. Available: https://www.threatq.com/

[94] Tines, "Tines - security orchestration, automation and response (SOAR) platform," 2020, last accessed 2020-10-20. [Online]. Available: https://www.tines.io/

[95] Cyware, "Virtual Cyber Fusion solutions," 2020, last accessed 2020-10-20. [Online]. Available: https://cyware.com/cyber-fusion-solutions

[96] NSA Cybersecurity, "WALKOFF," 2020, last accessed 2020-10-20. [Online]. Available: https://nsacyber.github.io/WALKOFF/

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.

[97] P. Clay, "A modern threat response framework," *Network 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," National Institute of Standards and Technology, Tech. Rep., 2016.

[99] C. Onwubiko and K. Ouazzane, "Soter: A playbook for cybersecurity incident management," *IEEE Transactions on Engineering Management*, 2020.

[100] D. Sparrell, "Cyber-safety in healthcare iot," in *2019 ITU Kaleidoscope: ICT for Health: Networks, Standards and Innovation (ITU K)*. IEEE, 2019, pp. 1–8.

[101] A. Harcourt, G. Christou, and S. Simpson, *Global Standard Setting in Internet Governance*. Oxford University Press, 2020.

[102] OASIS, *CACAO Playbook Requirements Version 1.0 - Committee Note Draft 01*, B. Jordan and A. Thomson, Eds. OASIS, 2020, last accessed 2020-11-15. [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," *arXiv preprint arXiv:2003.03663*, 2020.

[104] Cortex XSOAR, "Cortex XSOAR platform developer documentation - playbooks," 2020, last accessed 2020-12-15. [Online]. Available: https://xsoar.pan.dev/docs/playbooks/playbooks-overview

[105] DEMISTO, "Cortex XSOAR platform - content repository," 2020, last accessed 2020-12-15. [Online]. Available: https://github.com/demisto/content

[106] A. Iyer, *Security Orchestration For Dummies: Demisto Special Edition*. John Wiley & Sons, Inc., 2019.

[107] Johns Hopkins University Applied Physics Laboratory, "Integrated adaptive cyber defense (IACD) baseline reference architecture," 2016, last accessed 2020-10-15. [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," *Journal of Information 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 *International Command and Control Research Technology Symposium (ICCRTS), Annapolis, MD*, 2015.

[111] B. K. Dōne, 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 *Proceedings of the 2015 Workshop on Automated Decision Making for Active Cyber Defense*, 2015, pp. 1–1.

[113] Johns Hopkins University Applied Physics Laboratory, "Shareable automation and orchestration workflows for scoring, sharing, and responding to cyber indicators of compromise," 2020, last accessed 2020-10-15. [Online]. Available: https://www.iacdautomate.org/library

[114] ——, "Integrated adaptive cyber defense (IACD) orchestration thin specification," 2017, last accessed 2020-10-15. [Online]. Available: https://www.iacdautomate.org/library

[115] IACD, "Introduction to integrated adaptive cyber defense (IACD) playbooks," 2017, last accessed 2020-10-15. [Online]. Available: https://www.iacdautomate.org/library

[116] ——, "Types of content within an IACD playbook[s]," 2017, last accessed 2020-10-15. [Online]. Available: https://www.iacdautomate.org/s/IACD-Playbook-Content-Types.pdf

[117] OASIS, *Open Command and Control (OpenC2) Profile for Stateless Packet Filtering Version 1.0 - Committee Specification 01*, D. S. Joe Brule and A. Everett, Eds. OASIS, 2020, last accessed 2020-11-15. [Online]. Available: https://docs.oasis-open.org/openc2/oc2slpf/v1.0/cs01/oc2slpf-v1.0-cs01.html

[118] ——, *Specification for Transfer of OpenC2 Messages via HTTPS Version 1.0 - Committee Specification 01*, D. Lemire, Ed. OASIS, 2020, last accessed 2020-11-15. [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," *Computers & Security*, p. 101999, 2020.

[120] N. Kaloudi and J. Li, "The ai-based cyber threat landscape: A survey," *ACM Computing Surveys (CSUR)*, 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 *International Conference on Mobile, Secure, and Programmable Networking*. Springer, 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 *2017 International Conference On Cyber Situational Awareness, Data Analytics And Assessment (Cyber SA)*. IEEE, 2017, pp. 1–8.

[123] V. Mavroeidis and A. Jøsang, "Data-driven threat hunting using sysmon," in *Proceedings of the 2nd International Conference on Cryptography, Security and Privacy*, 2018, pp. 82–88.

[124] ATC Project, "RE&CT framework repository," 2020, last accessed 2020-10-01. [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 Communications Surveys & Tutorials*, vol. 22, no. 2, pp. 1191–1221, 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," *Digital Investigation*, vol. 6, pp. S57–S68, 2009.

[127] S. Garfinkel, "Digital forensics XML and the DFXML toolset," *Digital Investigation*, vol. 8, no. 3-4, pp. 161–174, 2012.

[128] E. Casey, G. Back, and S. Barnum, "Leveraging CybOX™ to standardize representation and exchange of digital forensic information," *Digital Investigation*, vol. 12, pp. S102–S110, 2015.

[129] C. Neiva, C. Lawson, T. Bussa, and G. Sadowski, "2019 market guide for security orchestration, automation and response solutions," Gartner, Tech. Rep., 2019.

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. The core research results show the importance of data quality aspects, incident response and threat sharing as application domains.

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

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.

D. Schlette, M. Caselli and G. Pernul, "A comparative Study on Cyber Threat Intelligence: The Security Incident Response Perspective," in IEEE Communications Surveys & Tutorials, doi: 10.1109/COMST.2021.3117338.