International Journal of Information Security (2024) 23:2951–2975
https://doi.org/10.1007/s10207-024-00869-1

REGULAR CONTRIBUTION

Swarm-intelligence for the modern ICT ecosystems

George Hatzivasilis[1] · Eftychia Lakka[1] · Manos Athanatos[1,2] · Sotiris Ioannidis[2] · Grigoris Kalogiannis[2] · Manolis Chatzimpyrros[2] · George Spanoudakis[2] · Spyros Papastergiou[3] · Stylianos Karagiannis[4] · Andreas Alexopoulos[5] · Dimitry Amelin[6] · Stephan Kiefer[6]

Accepted: 24 May 2024 / Published online: 18 June 2024

Abstract
Digitalization is continuing facilitating our daily lives. The world is interconnected as never before, bringing close people, businesses, or other organizations. However, hackers are also coming close. New business and operational models require the collection and processing of massive amounts of data in real-time, involving utilization of complex information systems, large supply-chains, personal devices, etc. These impose several advantages for adversaries on the one hand (e.g., poorly protected or monitored elements, slow fashion of security updates/upgrades in components that gain little attention, etc.), and many difficulties for defenders on the other hand (e.g., administrate large and complex systems with high dynamicity) in this cyber-security interplay. Impactful attacks on ICT systems, critical infrastructures, and supply networks, as well as cyber-warfare are deriving the necessity for more effective defensives. This paper presents a swarm-intelligence solution for incident handling and response. Cyber Threat Intelligence (CTI) is continuously integrated in the system (i.e., MISP, CVEs, STIX, etc.), and Artificial Intelligence (AI)/Machine Learning (ML) are incorporated in the risk assessment and event evaluation processes. Several incident handling and response sub-procedures are automated, improving effectiveness and decreasing response time. Information concerning identified malicious activity is circulated back to the community (i.e., via the MISP information sharing platform) in an open loop. The proposal is applied in the supply-chain of healthcare organizations in Europe (considering also EU data protection regulations). Nevertheless, it is a generic solution that can be applied in any domain.

Keywords Cyber threat intelligence · Risk assessment · Incident handling · Machine learning · Artificial intelligence · Swarm-inteligence · Healthcare · MISP · Drools · AutoML

List of Acronyms

AI	Artificial Intelligence
C&C	Command and Control
CAPEC	Common Attack Pattern Enumeration and Classification
CIL	Cumulative Impact Level
CIS	Center for Internet Security
CPE	Common Platform Enumeration
CSC	Critical Security Controls
CSIRT	Computer Security Incident Response Team
CTI	Cyber Threat Intelligence
CVD	Coordinated Vulnerability Disclosure
CVE	Common Vulnerabilities and Exposures
CVL	Cumulative Vulnerability Level
CVSS	Common Vulnerability Scoring System
DoS	Denial of Service
DNS	Domain Name System
EC	Event Calculus
ECVL	Entry's Chain Vulnerability Level
EHR	Electronic Health Record
ELK	Elasticsearch, Logstash, and Kibana
ENISA	European Union Agency for Cybersecurity
FVT	Forensics Visualization Toolkit
ICT	Information and Communications Technology
ICVL	Individual Chain Vulnerability Level
IDS	Intrusion Detection System
IDPS	Intrusion Detection and Prevention System
IOA	Indicator Of Attack
IOC	Indicator of Compromise
IPCI	Individual Propagated Chain Impact
IPVL	Individual Propagated Vulnerability Level
ISAC	Information Sharing and Analysis Center

Extended author information available on the last page of the article

IVL	Individual Vulnerability Level
MISP	Malware Information Sharing Platform
MitM	Man in the Middle
ML	Machine Learning
MTTResp	Mean Time To Response
MTTRest	Mean Time To Restore
NISTCSF	NIST cyber-security framework
NLP	Natural Language Processing
PA	Primary Agent
PIL	Propagated Impact Level
PVL	Propagated Vulnerability Level
R2L	Remote to Local
SA	Supervisory Agent
SEM	Security Event Management
SIM	Security Information Management
SIS	Smart Information Systems
SLA	Service Level Agreement
STIX	Structured Threat Information eXpression
TAXII	Trusted Automated eXchange of Indicator Information
TLS	Transport Layer Security
U2R	User to Root
UEBA	User and Entity Behavior Analytics

1 Introduction

The business landscape is changing towards the 4th Industrial Revolution [1, 2], and so the threat landscape is adopting as well [3–5]. The involved threat actors (both malicious and legitimate ones) are following this era of digitalization [6]. Threats are increasingly becoming sophisticated and sustained. Ransomware is currently the most prevalent emerging enterprise-wide business risk, with Deloitte estimating that cost for victims will reach the $265 billion by 2031 [7]. Social engineering is also advancing along with the exhibited malware capabilities. On the other hand, the risk management methodologies are also expanded to follow and, in some cases, try to surpass the wily strategies [8]. More-and-more organizations are enhancing their compliance and resilience elements. Fortune Business Insights forecasts that the global security market size will exceed the $376.32 billion in 2029, reaching a 13.4 CAGR [9]. The latest threat landscape reports from various organizations, like ENISA [10], NATO [11], IBM [12], McAfee, [13], and ESET [14], are also reflecting these aspects across the world and various business and operational sectors.

Attacks on the supply-chains and other logistics environments (e.g., on the USA energy sector [15]) are posing great threats for businesses and national security [16–18]. The ongoing Russia-Ukraine wars are also highlighting the potentials of cyber-warfare [19, 20] and the importance of building cyber-resilience in critical infrastructures [21–23].

Towards the achievement of these goals, defense strategies and mechanisms are evolving. Cyber Threat Intelligence (CTI) [24–26] is gathered and processed throughout the various phases of the Cyber Kill Chain (Reconnaissance, Weaponization, Delivery, Exploitation, Installation, Command and Control (C&C), and Actions on Objectives) [27, 28]. Thereupon, CTI sharing is emerging as a vital weapon in the defenders' arsenal to proactively cope with the increasing volume of malicious campaigns [29]. Automating the procedures for CTI consumption and sharing emerges as a new challenge for practitioners and researchers. The goal is to accomplish, in a timely fashion, situational awareness between the involved benign stakeholders by getting informed about potential threats to the current Information and Communications Technology (ICT) infrastructure setting. Automation in processing or implementation is desired by the involved entities for procedures like the caption of indicators of attack or compromise, preparation, response, sharing of CTI among trusted participants, etc. Thereupon, there are several solutions that provide semi-automated sharing of information, such as indicators with malicious/suspicious hashes or IPs. One main contribution has been provided by MITRE's STIX and TAXII in an attempt to advance the communal efforts towards a widely adopted protocol for CTI modelling and sharing [29, 30].

Cybersecurity in healthcare [31, 32] is of paramount importance as the industry increasingly relies on digital technology to manage patient records, conduct medical procedures, facilitate communication, etc. With the transition to Electronic Health Records (EHRs) and the proliferation of connected medical devices, protecting sensitive patient information from cyber-threats has become a critical concern. Healthcare organizations must implement robust security measures, including encryption, access controls, and regular security audits, to safeguard patient data from unauthorized access or breaches. As cyber-threats continue to evolve, maintaining a proactive and adaptive cybers-ecurity strategy is crucial in ensuring the integrity, confidentiality, and availability of healthcare information.

Regarding CTI, worldwide there are established Information Sharing and Analysis Centers (ISACs) specifically focused on healthcare [33]. Healthcare ISACs are organizations that facilitate the sharing of cybers-ecurity threat information and best practices within the healthcare industry. They serve as collaborative platforms where healthcare organizations, including hospitals, clinics, insurance providers, and other entities, can exchange information about cybers-ecurity threats, vulnerabilities, and protective measures. These ISACs play a crucial role in helping the healthcare industry stay vigilant against cyber-threats, which are of particular concern due to the sensitive nature of patient data.

They provide a forum for members to receive and disseminate timely and relevant threat intelligence, helping to enhance the overall cyber-security posture of the healthcare sector. By participating in an ISAC, healthcare organizations can benefit from collective intelligence and work together to address common security challenges.

This paper presents a swarm-intelligence solution, which was mainly developed under the EU funded project AI4HEALTHSEC. Security- and privacy-aware smart agents are continuously monitoring the systems under protection. The agents are self-organized as it concerns their main internal operation, while they also form a swarm-intelligence network where they can share security/privacy related information to safeguard the system-of-systems as a whole. Each individual organization deploys several Primary Agents that directly administrate their underlying local systems and networks, as well as one or more Supervisory Agents that collect knowledge from the Primary Agents and perform organization-wide decision making. Also, the Supervisory Agents from different organizations in a supply-chain or other collaborative environments, participate in a network and exchange high-level pieces of knowledge. CTI data could also be collected from other external collaborated communities or the Dark Web [34–36]. The agents utilize elements for Artificial Intelligence (AI) reasoning and Machine Learning (ML) inference to automate some of the underlying subprocedures (e.g., evaluation of ongoing events, process CTI resources with human-readable data, etc.) [35, 36].

The rest of the paper is organized as: Sect. 2 reviews the background theory and standardized methodologies for cyber security management. Section 3 presents the proposed solution for risk assessment and incident handling that was implemented under the EU funded project AI4HEALTHSEC. Section 4 details the application of AI4HEALTHSEC in a piloting environment for the protection of the healthcare organization FHG-IBMT. Section 5 provides discussions and directions for future works of modern CTI approaches. Finally, Sect. 6 concludes this work.

2 Background & related works

In 2019, the World Economic Forum's Global Risks Report stated cyber-attacks in its list for the top-ten most impactful global risks. Towards this direction, in the same year the Ponemon Institute reported that 90% of organizations supporting national critical infrastructures (i.e., manufacturing, industry, transportation, health, and energy) faced at least one assault within 2017–2019 that caused significant operational disruptions or data breaches [37]. Those are two of the many reports and studies that were held during the last decade on the landscape of cyber-security [7, 23]. Every year the volume and impact of malicious campaigns keeps increasing, revealing that such wily actions are forming an ever-growing threat for modern societies [38]. Therefore, concrete methodologies have been formed in order to cope with the various phases of the security lifecycle. The following subsections review the main methods and standards for incident handling.

2.1 Incident handling methodologies

In order to apply the best practices in preventing, handling, and managing all cyber security activities, it is first necessary to identify cyber security incidents. For this reason, many specific methodologies and frameworks for incident identification have been developed in the recent years.

Some consolidated procedures for security incident identification are defined in ISO/IEC 27035–1:2016 [39] and ISO/IEC 27035–2:2016 [40] standards. ISO/IEC 27035–1:2016 is the foundation of this multipart International Standard. It presents basic concepts and phases of information security incident management and combines these concepts with principles in a structured approach for detecting, reporting, assessing, and responding to incidents, while applying the lessons learnt. The principles given in ISO/IEC 27035–1:2016 are generic and intended to be applicable to all organizations, regardless of type, size, or nature. Organizations can adjust the guidance given in ISO/IEC 27035–1:2016 according to their type, size, and nature of business in relation to the information security risk situation. It is also applicable to external organizations providing information security incident management services. ISO/IEC 27035–2:2016 provides the guidelines to plan and prepare for incident response. The guidelines are based on the "Plan and Prepare" phase and the "Lessons Learned" phase of the "Information security incident management phases" model presented in ISO/IEC 27035-1.

The NIST cyber-security framework (NISTCSF) [41, 42] offers a quantitative and measurable risk reduction guide on how organizations can incorporate cyber security activities as part of their risk management process, including incident identification procedures. The framework provides guidance that is useful and applicable to any organization, therefore offers a common, consistent, and comparable set of guidelines and practices.

Another approach for incident identification and management relies on Computer Security Incident Response Team (CSIRT), whose main function is to react in a timely fashion to cyber security threats. The CSIRT will typically be called into action by a notification or a triggered event but can also be called into action by a relevant discovery while performing one of many passive services. The latter case may also include incident identification tasks. Frameworks for defining CSIRT services, roles, policies, standards, as well as procedures in case of incidents have been widely studied in literature [43–46].

The European Union Agency for Cybersecurity (ENISA) has provided a Good Practice Guide for Incident Management [47], which complements the existing set of ENISA guides that support CSIRT [48, 49]. The guide describes good practices and provides practical information and guidelines for the management of network and information security incidents with an emphasis on incident handling. In particular, it includes the identification of the incidents and its characteristics in the suggested procedures and handling process.Other existing standards that are also related with the various aspects of incident handling include:

- ISO/IEC 27039 [50]: Information technology—Security techniques—Selection, deployment, and operations of intrusion detection and prevention systems (IDPS),
- ISO/IEC 27041 [51]: Information technology—Security techniques—Guidance on assuring suitability and adequacy of incident investigative method,
- ISO/IEC 27042 [52]: Information technology—Security techniques—Guidelines for the analysis and interpretation of digital evidence,
- CRR Supplemental Resource Guide [53]: Volume 5 Incident Management Version 1.1, Carnegie Mellon University,
- ITU-T X.1216 Telecommunication Standardization Sector of ITU (09/2020) Series X [54]: Data Networks, Open System Communications and Security Cyberspace security—Cybersecurity Requirements for collection and preservation of cyber-security incident evidence.
- SANS Institute [55]: Computer Security Incident Handling: Step by Step, a Survival Guide for Computer Security Incident Handling.

2.2 Analysis and comparison

Considering all the above, many well-documented methodologies that describe the security incident response process, have already been proposed and applied in the healthcare domain. As mentioned before, the major aim of these strategies is to analyze a procedure for rapid detection of incidents, along with minimizing the effects, mitigating the causes, and restoring the affected resources. In fact, the popular Incident Handling recommendations proposed by ENISA [47–49], NIST [42], ISO/IEC 27035–1 [39], and CSIRT and CERT/CC [44, 45].

As shown already above, all approaches share common characteristics, and it seems possible to derive a general methodology which would cover the entire procedure by the conjunction of the practices introduced by the various sources. However, these methods might exhibit deviations concerning the definition and coverage of the terminology used.

The AI4HEALTHSEC project team, after carefully investigating the relevant methodologies, defined and implemented a practical methodology described below. To help with the visualization of the common points between the basic methodologies and to show the basis of the AI4HEALTHSEC proposed methodology, Table 1 has been created.

Initiality, most methodologies start with the cyber-security preparation of the organization to establish an incident response capability. This involves developing and implementing an incident response plan, setting up an incident response team, and providing training and awareness for staff. Preparation is key to effective incident management, as it ensures that the necessary tools, roles, and procedures are in place before an incident occurs.

Thereupon, the deployed defenses are starting to identify and detect potential security incidents. This phase ordinarily includes the monitoring of systems and networks for signs of an incident, the establishment of detection mechanisms, and the procedures for reporting potential incidents. Timely detection is crucial for minimizing the impact of security incidents.

Once an incident is detected, it needs to be assessed to understand its nature and scope. These phases involve triaging events, evaluating the severity of the incident, determining its impact, and deciding on the appropriate course of action. This step is critical for prioritizing and responding to incidents effectively.

After the detection and assessment, the core incident response is performed. The organization takes action to contain, eradicate, and recover from the incident. This may include measures to limit the spread of the incident, remove the cause of the incident, and restore affected systems or data. The response actions are based on the type and severity of the incident.

Finally, when managing the incident, the organization should review and analyze it, as well as the response to it. This phase aims to identify lessons learned, improve the incident response plan, and enhance security measures to prevent future incidents. It is a crucial step for continuous improvement in incident management and overall security posture.

The principles [46] that the state-of-the-art methodology should have, are: (1) Preparations and receiving, (2) Triage, (3) Analysis, (4) Response, and (5) Post-incident activities.

2.3 AI4HEALTHSEC incident handling phases

Taking into account the previous analysis, a new derivative methodology can be developed by adapting the NIST methodology to the swarm-intelligence architecture scheme of AI4HEALTHSEC [42]. Hence, the incident handling framework of AI4HEALTHSEC consists of four steps, as they are proposed by NIST [42] (see Fig. 1):

Table 1 Common characteristics in incident response approaches

Suggested Principles	NIST	SANS	ENISA	CERT/CC	ISO 27035–1
Preparation and receiving	Preparation	Preparation	Receiving incident reports	Detecting and reporting	Plan and prepare
Triage	Detection and analysis	Detection and analysis	Incident evaluation	Triage	Detection and reporting
Analysis				Analysis	Assessment and decision
Response	Containment, eradication, and recovery	Containment	Actions	Incident response	Responses
		Eradication			
		Recovery			
Post-incident activity	Post-incident activity	Post-incident activity			Lessons learnt

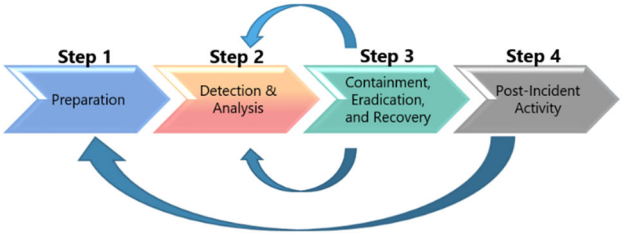

Fig. 1 NIST & AI4HEALTHSEC Incident Respond Life Cycle

- **Preparation (Step 1):** It contains the steps that are taken before an incident occurs, such as training, writing incident response policies and procedures, and providing tools such as laptops with sniffing software, crossover cables, original OS media, removable drives, etc. In fact, preparation should include anything that may be required to handle an incident or will make incident response faster and more effective.
- **Detection and analysis (Step 2):** It is the phase in which events are analyzed in order to determine whether these events might comprise a security incident (triage principles are included in this step).
- **Containment, eradication and recovery (Step 3):** The containment phase of incident response is the point at which the incident response team attempts to keep further damage from occurring as a result of the incident (i.e., taking a system off the network, isolating traffic, powering off the system, etc.). The eradication phase involves the process of understanding the cause of the incident, so that the system can be reliably cleaned and ultimately restored to operational status later in the recovery phase. The recovery phase involves cautiously restoring the system or systems to operational status.
- **Post-incident activity (step 4):** It includes the creation of a follow-up report, which each incident response team should evolve to reflect new threats, improved technology, and lessons learned aiming to reduce the probability of a similar incident happening again and to improve incident handling procedures.

3 The AI4HEALTHSEC approach

As aforementioned, AI4HEALTHSEC forms a swarm-intelligence framework that can protect the systems of distinct organizations and their supply-chains. This is implemented in the form of two main procedures for risk assessment and incident handling, respectively.

The risk assessment elements are based on the MITIGATE platform (see details below). The system: (1) records the organizational assets, (2) discloses their known vulnerabilities and threats (e.g., from the Common Vulnerabilities and

Exposures (CVE) [56] or other databases [57]), and (3) estimates the levels of individual and cumulative risks of the evaluated components. ML is used in order to enable the processing of human language and the automatic ingestion of information from relevant resources (e.g., [35, 36]). This risk assessment procedure with MITIGATE is presented in [58, 59] and it is not covered further in the current paper. Nevertheless, an outline of MITIGATE is provided in the sub-Sect. 3.1 for completion.

This study details the incident handling process. Smart agents are collecting data from the monitored devices and networks, exchange fruitful information or gather CTI from open sources (similarly with the abovementioned risk assessment elements), and build knowledge concerning the overall operational environment. This "CTI of Things" concept [60] permits the secure administration of the whole ecosystem, with AI/ML automating several of these steps and mitigating some type of zero-day attacks (e.g., [61–63]).

3.1 Outline of risk assessment with MITIGATE

The AI4HEALTHSEC Risk Assessment methodology that is implemented with MITIGATE [58, 59], comprises five systematic phases for managing cyber-security risks in healthcare operations. It outlines a process that guides organizations in comprehending and controlling risks, focusing on crucial aspects like the healthcare supply-chain, assessing critical assets, threat profiling, risk evaluation, and prioritizing controls. By following MITIGATE, healthcare institutions can gain insights into individual and cascading risks, enabling them to implement appropriate controls for a secure and robust healthcare ICT infrastructure. This methodology provides a structured approach for healthcare organizations to enhance overall cyber-security. MITIGATE aligns with established risk management standards such as ISO-31000:2018 [64] and ISMS standard ISO-27001:2013 [65], as well as security standards like the Center for Internet Security (CIS) Critical Security Controls (CSC) (CIS_CSC) [66] and Common Vulnerability Scoring System (CVSS) [67]. Additionally, it incorporates recognized security best practices and methodologies, including CVSS 4.0 for vulnerability assessment [67], the Common Platform Enumeration (CPE) [68] for asset mapping, Common Attack Pattern Enumeration and Classification (CAPEC) for threat identification [69], Coordinated Vulnerability Disclosure (CVD) for vulnerability identification [70], consideration of healthcare sector assets, and the integration of suitable ML models [35], making it applicable across various supply-chain domains.

These five risk assessment phases are:

- **Phase 1:** The initial phase involves establishing the overall scope of risk management within the healthcare organization and its supply-chain context. Active participation of key stakeholders, such as operations managers, IT and security managers, and business analysts, is essential to precisely define this scope. The outcomes of this phase are: 91) the specification of the cyber security risk management scope, and (2) the identification of the organizational context, which encompasses both internal and external facets, as well as outlining the risk management strategy.
- **Phase 2:** Following the determination of risk management scope and context, the subsequent phase entails an analysis of the healthcare organization's environment and the existing healthcare ecosystem it operates in. The outcomes of this phase are: (1) a list of all available healthcare services of the organization, classified based on their criticality level, (2) the list of the underlying assets with their CPE identification, the services that each asset participates in, and the dependency and criticality values for each of the assets and services, and (3) the identification of healthcare ecosystem dependencies as well as the production of a Healthcare Ecosystem Dependency Graph.
- **Phase 3:** This phase aims to identify all potential individual risks that may affect the assessed assets. It comprises four steps that provide a comprehensive insight into vulnerabilities and threats relevant to these risks. This process aids in analyzing risks from a holistic perspective of the healthcare ecosystem and formulating appropriate control measures to mitigate the identified risks. The outcomes of this phase are: (1) the identification of the individual threats along with their occurrence level (ranging from very high to very low) based on the links of assets' CPE on the CAPEC list, the related CVEs and their risk scores via the CVSS, (2) the production of a related asset vulnerability inventory and the identification of the Individual Vulnerability Levels (IVLs) for all of the entries, (3) a list of impacts per relevant asset that operates in the context of each identified healthcare service, and (4) a calculated risk value for each vulnerability of each recorded asset of each recorded service.
- **Phase 4:** In this phase, the methodology focuses on recognizing and evaluating cascading risks associated with assets and services in the healthcare supply-chain. RA4Health specifically takes into consideration vulnerabilities that could be exploited due to interdependencies among assets. The outcomes of this phase are: (1) the estimation of the Individual Chain Vulnerability Level (ICVL), the Entry's Chain Vulnerability Level (ECVL), the Cumulative Vulnerability Level (CVL), and the construction of vulnerability dependency chains, (2) the generated propagated vulnerability chains from the exposure of a specific vulnerability in an asset entry point, the estimation of the Individual Propagated Vulnerability Levels (IPVLs) and the calculation of the Propagated Vulnerability Levels (PVLs), (3) the Cumulative Impact Level (CIL) for each vulnerability chain, iv) the estimation of the

Individual Propagated Chain Impact (IPCI) and the Propagated Impact Level (PIL), v) the Cumulative Risk level of each target asset, and (4) the Propagated Risk level of each asset considered as an entry point of an attack for a specific threat.

- **Phase 5:** The final phase of the MITIGATE methodology implements an effective risk management strategy to address both identified individual and cascading risks, ensuring a resilient healthcare service delivery [71]. This involves a decision-making process, based on existing risks, to select the most suitable strategy and recommended controls within various constraints (e.g., healthcare context, budget, and resource availability). The phase also revisits vulnerabilities and threats identified in prior stages to pinpoint appropriate control measures. Additionally, it aligns with established standards to demonstrate the broader applicability of these controls. The resulting guidance aids the healthcare entity in implementing the identified controls, managing risks, and enhancing overall security and resilience. The outcome of this phase and the final result of the risk assessment procedure is a control register. It establishes a connection between the identified control, the vulnerability, threat, and assets, enabling a clear traceability path from control to threats and vulnerabilities.

3.2 Overview of the swarm-intelligence network for incident handling

The AI4HEALTHSEC system is composed of a set of smart agents. Each organization has at least one Supervisory Agent (SA), which manages the underlying systems and networks, exchanges information with SAs from other collaborative organizations, and collects CTI from open sources. Then, within each organization and under the relevant SA, there can be several Primary Agents (PAs) that directly audit systems and networks. For example, there can be specialized PAs that monitor traffic in a networking gateway or the operation of nodes (e.g., computers, mobile or medical devices, etc.). The main incident handling operations for a SA are mainly administrated by the Assurance Platform [71] component and for a PA by the Metadon [73] component (see the following paragraphs for details).

Both agents are self-organized, in the sense that they have the knowledge to handle some types of events by their own. They also exchange information to achieve collaborative tasks. The SA is higher in this hierarchy. It collects information from all sources and has the holistic view of the organization's system. It is also the main interaction point with the human operator for incident handling, who can also access the PAs in order to perform direct human-driven analysis (if required).

The formal aspects of Artificial Intelligence (AI) are based on the Event Calculus (EC) [74] and the implementation of this reasoning behavior is developed with the Drools reasoning engine [75]. Apart from a simple rule-based logic, the reasoning process can be enhanced with ML features that further evaluate ongoing events and feed the results back to the AI reasoning. ML and AutoML modules are built for this purpose, which are based on the Keras and AutoKeras system [76], respectively. The capturing of ongoing events from the running system is based either on customized event captors and/or the Elasticsearch, Logstash, and Kibana (ELK) Beats [77]. ELK may also act as an internal knowledge base for the agents within an organization. Also, the agents can interchange messages between them and other AI4HEALTHSEC elements (e.g., the risk assessment components) via a Kafka Broker [78]. The privacy module, implemented with CHIMERA [79], can be also executed to anonymize messages' content with personal data. CTI collection within the organization and exchange with other entities or communities is developed in the Malware Information Sharing Platform (MISP) [80]. Finally, the user can access the AI4HEALTHSEC platform via a unified web interface and the Forensics Visualization Toolkit (FVT) [81], where he/she can find the details for each recorded event and its analysis elements. Figure 2 illustrates the main components for the installation of AI4HEALTHSEC in a single organization with many local PAs (Metadon) and one SA (Assurance Platform) at the backend.

3.3 Preparation

During the initial deployment of the platform to an organization, the audited system elements are recorded in the Assets Inventory (e.g., asset ID, vendor, version, correlation with other assets, etc.). Then, the risk assessment analysis with MITIGATE discloses the currently known vulnerabilities of these component based on the latest CTI, which is collected from various resources, like general CVEs, sectorial repositories (e.g., with vulnerabilities for medical devices), and Dark Web. Usually, repositories contain machine readable data, and their consumption can be facilitated with STIX/TAXII compatible services [82]. Human-readable sources can be also incorporated (e.g., CVEs' description part or cybersecurity blogs) with the help of the Natural Language Processing (NLP) module [35, 36].

Thereupon, the organization lists the most critical threats and prioritize the defenses. The incident handling elements are applied, and the smart agents are configured accordingly. Capturing mechanisms are deployed to continuously sending data to the agents, which will evaluate all events based on the implemented Artificial Intelligence (AI) rule-based logic. If further Machine Learning (ML) evaluations are required, the initial construction of datasets and training activities are

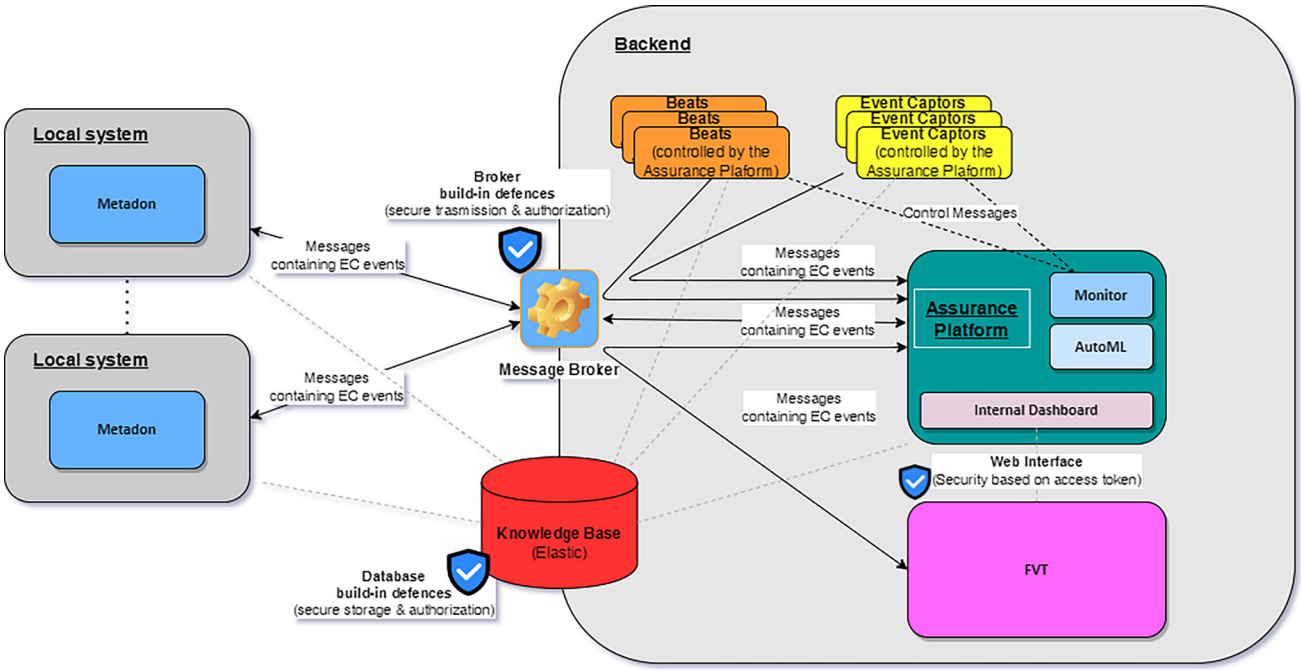

Fig. 2 Interactions between the Assurance platform and other internal components

performed in this step. Afterwards, the platform becomes ready for the monitoring and management of the runtime operation of the system.

3.4 Detection and analysis

The phase of Detection and Analysis incorporates automated processing tools and intelligent operations conducted by the AI4HEALTHSEC agents. Additionally, the Analysis phase involves the examination and analysis performed by the system operator situated at the SA's side.

3.4.1 Gathering of information

The detection phase involves utilizing various techniques and tools to capture and identify cyber incidents.Information sources include log management tools, antivirus software, intrusion detection systems, network monitoring tools, and vulnerability scanners, among others. Logging standards and procedures play a crucial role in ensuring the collection of information from logs and security software.

Both agent types (PAs and SAs) implement mechanisms that can deploy Customized Event Captors to gather and fetch information from the abovementioned information sources. Additionally, the Assurance Platform of the SA can utilize ELK Beats and their capabilities which are popular and widely known for the experts in the field.

The AI4HEALTHSEC platform prioritizes effective log management to handle vast amounts of data, enabling security analysts to conduct queries and correlations that aid in the detection phase. Specifically, it emphasizes Security Information Management (SIM) and Security Event Management (SEM). These procedures utilize a standardized taxonomy for security events and incidents, along with a shared information model aligned with governmental and industrial standards [83, 84].

Various methods are employed for log collection connectors, which include data ingestion to the manager through HTTP, TCP, and syslog standards, as well as monitoring specific log/text files. A transformation process is implemented to convert input formats from a domain-specific language into a standardized and uniform field taxonomy. Elasticsearch [85] serves as the storage solution for the data. A web-based user interface facilitates security analytics and enables the execution of queries using a high-level query language. Events and alerts are generated when abnormal system behaviors or operations occur. To enhance search queries, visual analytics techniques are utilized to visually represent patterns such as unusual network behavior, unexpected configuration changes, and anomalous account activities, aiding human operators in effectively identifying cyber incidents.

Once security- and privacy-related data has been collected, primarily using Elasticsearch, three types of tools are employed to process this information during the detection phase of the AI4HEALTHSEC methodology. These tools

Table 2 Captured malicious activities

Malicious activity	
Brute force attempts	Attempt of log tampering
Multiple failed authentications to different accounts by same IP	Sensitive file permissions change
Access to the same account from different IPs	Sensitive group change
Password change after new IP access	Suspicious user elevation
Access from/to blacklisted IP	Suspicious number of VM/docker activity
Possible man in the middle (MitM)	High NXDomain enumeration
Too many errors 404, 403, 500, 501	High reverse DNS enumeration
Exposed endpoints used from public IP	Failed access to host but successful access to system
Detection of accesses to sensitive protocols	Multiple users email forwarding to same destination
Anomalous traffic detection (leveraging from intrusion detection systems),	Anomalous sensitive service execution
Commonly abused URLs	User created and deleted within 10 minutes
Distributed brute force attempt	Process executed from binary hidden in Base64 encoded file

consist of: (1) an Intrusion Detection System (IDS) based on Snort [86], (2) a Data Fusion module and CHIMERA [79] for privacy protection, and (3) Metadon [73] for PAs or the Continuous Monitoring and Assurance Platform [71] for SAs. Snort fulfills the primary functionality of the IDS. The Data Fusion module stores pertinent information for this phase, handles intricate data transformations, and applies semantic annotations to facilitate knowledge sharing and reuse. CHIMERA may apply additional anonymization transformations in cases where personal data are processed. Metadon and the Assurance Platform utilizes all the gathered information to validate the compliance of the operational environment with established security policies/strategies, identifying any violations or potential cyber incidents.

The Table 2 summarizes the main type of malicious activities that can be detected by the AI4HEALTHSEC components.

In an indicative use case, the detection process can be exemplified by employing a security appliance, such as a firewall, that generates logs. These logs are then collected and parsed into a standardized format using a syslog server. The resulting data is transmitted to the log storage, which utilizes Elasticsearch. Subsequently, users have the ability to define queries on the collected logs using a language that is not specific to any particular domain. Correlation tasks can be performed by leveraging the data. By scheduling queries, it becomes possible to trigger alerts, notifying users through email, initiating a service, or executing a shell script when specific conditions are met. Any log source can be utilized as input, encompassing auditing logs, firewall logs, router logs, DNS traffic, or network flows. The system supports multiple data formats, including CSV, JSON, and raw data in legacy formats, accommodating nearly all types of data.

3.4.2 Reasoning evaluation and response

After capturing and fetching events from the monitored systems and networks, the smart agents can process all these pieces of information and reason about their legitimacy and incident handling policies.

The core reasoning procedures are implemented in a rule-based logic. Event Calculus (EC) is mainly used in order to model the reasoning behaviors and provide the theoretic foundations for the formal verification of the evaluation process. EC is implemented in the rule engine Drools and the AI procedures are implemented in JAVA. Set of rules are defined, with each rule-set evaluating a specific type of security or privacy incident.

Thereupon, the user can deploy Assessment Profiles. Each profile contains information about the deployment of the related capturing mechanisms to specified system assets, as well as which rule-sets will be used for the evaluation of the collected events. Each incoming event triggers the reasoning behavior to evaluate to situation. The user can review the results and the whole process via a user-friendly web interface. Moreover, these AI procedures can execute automated response actions, which are defined in the deployed rule-sets, to response and mitigate ongoing malicious operations (see the following paragraphs).

3.4.3 Incorporation of reasoning with ML evaluation

While rule-based AI procedures are quite powerful and can cope with a wide range of problematic situations, there are complex problems that cannot be easily expressed and modeled with them. Machine Learning (ML) elements are also required in order to provide evaluation of complex wily activities as well as anomaly detection.

Therefore, enhanced reasoning behaviors are developed for AI4HEALTHSEC, where the AI procedures can ask for ML components to evaluate series of events. When a relevant rule-set is triggered by one or more incoming events, one of the underlying rules can send a message to an ML component asking for evaluation. The ML component responds with the evaluation result, which is processed by one or more rules of the rule-set.

Context-wise, usually PAs process more specialized and technical datasets (e.g., specific network monitoring, devices profiling, etc.), while SAs process datasets for more common services and functionality that is found in backend infrastructure. In the current prototype of AI4HEALTHSEC, ML and AutoML modules [62] are embodied by the agents based on Keras and AutoKeras solutions. The ML solution is mainly used by the PAs and concentrates in datasets for local systems and networking traffic. The AutoML module is mostly used by the SAs. This solution is also better fitted for users who have no or low expertise in ML and are not aware of how to choose a proper ML algorithm for their problem.

Although these two ML components were selected for the proof-of-concept implementation of AI4HEALTHSEC, the overall approach is generic enough and can easily incorporate other ML solutions as well.

3.4.4 Human-driven analysis

The proposed solution relies not only on the accuracy of system indicators or precursors but also on human judgment in certain cases [87]. Indicators may not always signify an actual incident, and the complexity of interconnected ecosystems results in varying types and quantities of indicators, whether user-defined or not. Identifying genuine security incidents from the multitude of indicators and associated events can be a challenging task. To facilitate these processes and examine different signs and events within the system, incident handlers utilize tools such as the Forensics Visualization Toolkit (FVT) [81].

During the Incident Analysis process, a security expert examines incoming events by investigating the system's status through different visualizations and dashboards. These visualizations provide diverse perspectives based on potential event/incident correlations. The exploration schemas can be pre-configured for specific event types or customized based on the investigator's requirements. Various visualizations and dashboard schemas correspond to different system metrics (e.g., device monitoring, network monitoring) or event-related information (e.g., incident entry point, potentially affected nodes). The security expert can also request and analyze a variety of data from individual event processing and analysis services, leveraging knowledge from all available sources. This knowledge includes previously identified incidents, their semantic descriptions in the form of Data Reference Models, and correlations with ongoing investigations or historical events. Filtering and comparing this data facilitate network-system profiling, understanding normal behaviors, and discovering new attack patterns.

Throughout the incident analysis process, the security expert can explore and modify the incident profile, which is documented and stored in knowledge databases for future reference. Incidents can be prioritized based on their functional and informational impact, as well as the system's recoverability level. The incident prioritization process may involve user-driven or automated simulations and AI-based recommendations. Once an incident is identified as a potential attack, the initial level of incident response is initiated, involving the notification of all affected parties. This notification process includes generating alerts and identifying the recipients who should be notified based on established security policy conditions (e.g., notification period, level of information to be shared).

Moreover, this stage can involve an interplay of the incident handling procedure and the risk assessment one. Therefore, the analyst can utilize the offerings of MITIGATE [58, 59] to further evaluate the potential side-affects. For example, an attacker may have infiltrated the system from some point and the attack is detected at some time-point in a specific part of the system. The expert can run the attack simulation module in an attempt to examine the defined Chain Vulnerability paths (already defined during the initial risk assessment phase) and how the vulnerability may have been propagated in the system, trying the trace backward and forward the hacker and mitigating the side-affects.

When the incident has been handled, the analyst can determine the overall feedback, updating and re-performing the risk assessment to estimate the new protection level. This may include the update of the vulnerability database, the vulnerability paths, and the affected risk levels [71]. Moreover, the user could annotate the relevant events, indicating also potential false positives/negatives, and re-train the ML models that are utilized in vulnerable assessments and incident/anomaly identification and are briefly described in the Sect. 4.4.

3.4.5 Containment, eradication, and recovery

Containment is the phase where the incident is controlled, either through the response team's actions or automated processes, by isolating network and asset variations that could be affected by the original attack. This phase also involves predicting potential future targets that the incident may impact. Attack propagation graphs are valuable in capturing and extracting the necessary information to guide the isolation tasks (link to MITIGATE analysis). The goal of containment is to limit the damage caused by the incident and subsequently eradicate or remove any malicious artifacts, followed by system recovery as outlined in NIST 800–61 guidelines [88, 89]. In essence, containment aims to minimize the incident's impact and prevent further contamination of the system. During the containment process, evidence must be collected and analyzed. The eradication phase involves removing suspicious artifacts based on Indicators of Compromise (IOCs), while the recovery phase focuses on restoring the system to

normal operation and continuously monitoring its state after the incident [90].

Containment is an interdependent task that encompasses security, policy, and network management [91]. Security operations handle compromised devices, assets, or attacks by transitioning to a secure domain using security controllers [92]. Containment methods can range from simple actions like disconnecting a network cable or shutting down processes to more complex measures such as isolating compromised machines through Domain Name System (DNS) or firewall rule changes [93]. The containment phase relies heavily on the information gathered during the detection and analysis phase, which is used to identify and define IOCs for system neutralization.

The specific nature of attacks determines the technical aspects of containment, considering attributes like malware, rootkits, Denial of Service (DoS), asset loss, data theft, unauthorized access, and misuse of assets [94]. IOCs play a crucial role in matching them to the incident and proceeding with containment and system isolation for further analysis. Research has focused on containing malware attacks, proposing strict or flexible rules to restrict the attacks. The chosen containment strategy may involve complete isolation, filtering, or emulation procedures. Therefore, containment strategies vary based on the incident type, and specific criteria have been proposed to determine the appropriate approach. To verify and validate the compromised host, incident handlers should consider containment-specific activities for identifying the attack host.

Within the context of this project, containment processes encompass configuring and updating firewall rules and isolating compromised machines by changing DNS settings and restricting them to a virtual network. This process is triggered by event matching and correlation, where specific rule-sets are monitored. The alerting system then triggers commands to update firewall rules, effectively redirecting the compromised device to a different network. The compromised device remains isolated while retaining Internet connectivity and accessing a different network environment. It operates under monitored actions to gather data that will aid the eradication phase.

Critical post-containment actions include analyzing and extracting IOCs, focusing on containment and preparing for eradication. IOCs can include virus signatures, changes in file integrity within system registries, inbound and outbound network traffic, or previously reported malicious domain names.

The actions taken to eradicate the effects of an incident depend on the type of attack. For instance, in the case of malware or ransomware attacks, it is necessary to delete the malicious files, restore file integrity changes, and reverse any registry modifications. In a DoS attack, updating rule-sets in firewalls and intrusion detection/prevention systems and implementing new technologies or tasks to prevent future similar attacks are crucial. If the incident involves a rootkit, the process involves identifying and recovering the original system image from previous backups or reinstalling the infected OS from scratch. In all cases, it is essential to validate that no malicious artifacts, processes, or configurations related to the incident are still present [89].

Once the malicious code is removed and the system services are restored, it is important to perform iterative vulnerability assessments to ensure that security and configuration flaws have been addressed. A review should be created based on this vulnerability assessment, ensuring a successful recovery phase that builds upon successful containment and eradication efforts. Understanding the full extent of the damage is crucial, and various logs such as system logs, IDS logs, configuration logs, and incident documentation can be utilized to support this endeavor.

Concerning recovery, two main methods exist for: backward recovery and forward recovery [95]. Backward recovery involves restoring a system to a previous state that is known to be uncompromised. In such cases, backups created close to the incident time are used to minimize restoration time. Backward recovery is particularly useful when the extent of the damage is difficult to determine or when there is a lack of confidence, especially during the eradication phase. On the other hand, forward recovery involves undoing tasks based on logged information. By examining system logs that have recorded malicious activities, attempts can be made to reverse the effects caused by those activities.

3.4.6 Incident-related information sharing

3.4.6.1 Outline As part of the core functionality that is performed by the Supervisory Agent (SA), the Assurance Platform facilitates the information sharing operations of the incident handling process. These include internal communications within a healthcare organization, as well as external interactions with other entities. Internal communications involve: 91) the exchange information with the underlying Primary Agents (PAs) (via relevant Metadon instances), (2) fusion of events from various resources and potential reasoning (by the Assurance Platform itself) concerning the overall status, and (3) notification of the backend user/analyst regarding high level events and incidents with high severity (through the FVT component of the Unified Dashboard). External interactions involve: (1) the exchange of information with other Supervisory Agents (instances of the Assurance Platform in other organizations) concerning high level events or recorded incidents on interconnected assets (i.e., assets that are used in common or participate in a collaborative service between the organizations), as well as (2) the filtering of captured events that are related with Cyber Threat Intelligence (CTI) (e.g., identified malicious

IPs, domains, and spam emails or signatures of malicious code) and are disseminated in the MISP AI4HEALTHSEC.

3.4.6.2 Internal information sharing between the different nodes within the same organization As mentioned before, the Assurance Platform at the backend (acting as the SA) collects information from the underlying Metadon instances at the edge systems (acting like the PA) and orchestrates the internal swarm-intelligence functionality (between the SA of an organization and its underlying PAs).

Regarding the log collection by using PAs, the Metadon agents collect data from multiple resources and distribute the data either to the Metadon or to different nodes.

Metadon agents can send data between them, meaning that in whichever system the agents are deployed the logs can be retrieved there or the opposite. Furthermore, the Metadon agents retrieve logs that can be sent to the Metadon service and store the indexes appropriately there. Therefore, the Metadon agents can be used for storing and managing the logs to the cloud service of AI4HEALTHSEC or to distribute to other third parties by using a peer-to-peer connection between the agents. The communication is encrypted, and the connection is using regular HTTP protocol in order to bypass security limitations and maintain flexibility on the deployment.

Listeners are used to inserting the logs collected on the AI4HEALTHSEC components into the Metadon's database. Configuring a listener is basically defining a channel of HTTP or Kafka type, where the components will put through the data collected on each component. Moreover, all data will be normalized according to the pre-defined rules, and then inserted into Metadon's database.

The formal aspects of the reasoning behavior of the Platform are based on Event Calculus (EC), which is implemented with the Drools reasoning engine and the Java general programming language. The Platform exchanges messages with the rest AI4HEALTHSEC components via a broker. The content of the messages includes EC events that represent the security/privacy events/incidents that have been identified.

Thereupon, the SA processes and reasons about these pieces of knowledge and can perform a series of actions:

- Store the information in the Knowledge Base
- Send information for selected type of events (e.g., incidents with high severity, metrics, etc.) to the Unified Dashboard
- Send information for selected type of events concerning Cyber Threat Intelligence (CTI) to the MISP AI4HEALTHSEC
- Perform pre-defined responsive actions (e.g., send email to the system administrator for an on-going attack)

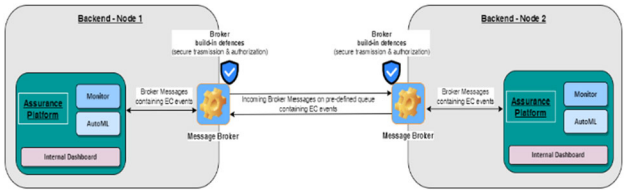

Fig. 3 Information sharing between different organizations of the swarm network

- Command Metadon instances to perform pre-defined responsive actions to their local system (e.g., increase security level proactively due to attacks and/or anomalies that are currently observed in other system areas)
- Inform collaborating entities (other nodes of the swarm-intelligence network) regarding on-going attacks/anomalies on their interconnected assets.

The Fig. 2 illustrates these interactions of the internal components within an organization.

3.4.6.3 External information sharing with other organizations in the swarm network This subsection documents the last action of sharing information between the different nodes of the swarm-intelligence network. This is performed in the form of information exchange between the different SAs. Each node/entity has installed in its backend a relevant instance of the Assurance Platform, performing the main reasoning behavior of the SA. The Assurance Platform instances communicate by exchanging messages via Kafka brokers. Each instance has its own broker (which also facilitates the above-mentioned internal communications) and 'listens' to a queue, where other instances can send messages containing EC events. Then, the processing of knowledge is performed as described before and the aforementioned actions can be executed/triggered (i.e., store information, perform responsive actions, command PAs, notify user, or share information to the swarm-intelligence network). The information sharing between different nodes is depicted in Fig. 3.

The information in-transit is protected with Transport Layer Security (TLS). Self-signed certificates have been produced for each instance and the system administrator installs the certificates of the trusted entities in the Assurance Platform instance of his/her organization. The instances listen in pre-defined broker queues for incoming messages from those trusted nodes, applying also authorization properties. Information about the content of transmitted messages cannot be disclosed by adversaries and captured messages cannot be retransmitted (replay attack). Moreover, non-trusted entities cannot send information (the authentication fails, and their messages are dropped).

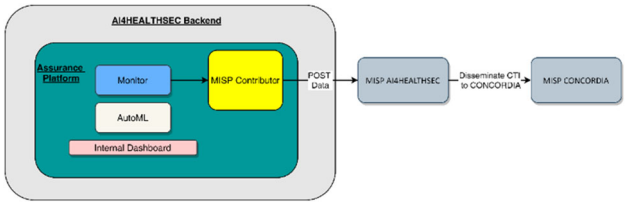

Fig. 4 Information sharing between AI4HEALTHSEC and CONCORDIA community

3.4.6.4 Information sharing to external stakeholders and communities AI4HEALTHSEC will also support information sharing with external stakeholders. Specifically, the human operator of a SA will be able to collect CTI data and disseminate it in the MISP AI4HEALTHSEC, which is responsible to provide functionalities related to CTI exchange, integration with the EU-funded multi-disciplinary research and innovation project CONCORDIA [96]. The SA deploys a MISP contributor service which gathers CTI data from the underlying system and the final message can be sent to the MISP AI4HEALTHSEC and therefore to the CONCORDIA's community.

CONCORDIA is part of the Cybersecurity Competence Network of EU. CONCORDIA strives for excellence and leadership in technology, processes, and services to create a user-centered EU-integrated cyber-security ecosystem, aiming to promote digital sovereignty in Europe. The cybersecurity community includes CTI sharing organizations, such as ER-ISAC, EE-ISAC, EA-ISAC, the EU CSIRTs Network, ENISA, and EUROPOL [96]. One of the project's outcomes includes a service to automatically detect a threat in the network using indicators of compromise provided via an instance of the MISP platform.

As mentioned in the previous subsections, the Assurance Platform collects information concerning the underlying system of a healthcare organization, as well as the swarm-intelligence network. Then, the Platform can reason about the collected pieces of knowledge and perform related actions. One such action is to filter the recorded events and notify the MISP Contributor. This module provides the corresponding functionality that the Assurance Platform calls every time a relevant event is identified. Figure 4 illustrates the information sharing between AI4HEALTHSEC and the CONCORDIA community.

4 Demonstration in FHG-IBMT

4.1 Piloting system description

FHG-IBMT is the main organization that is examined under this risk assessment process. It collects and maintains important biorepositories and provides human biomaterial for research purposes. FHG-IBMT also collects and stores human samples from specific cohorts of donors to monitor people's exposure to contaminants in the environment on behalf of the German Environment Agency UBA. Users of the system that access the main FHG-IBMT web-application via Internet.

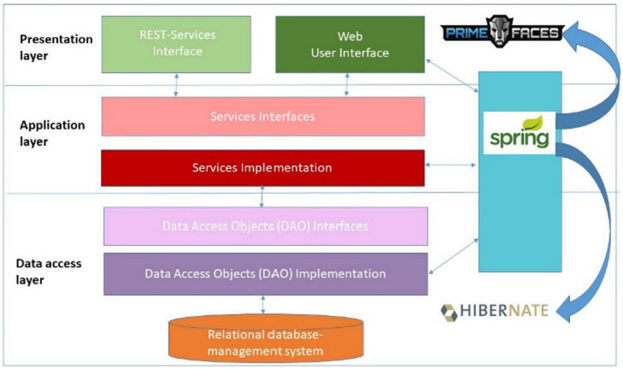

Fig. 5 Main architecture of the piloting testbed of FHG-IBMT

The Fig. 5 illustrates the main system architecture of the piloting testbed that was provided by FHG-IBMT and was utilized for the evaluation of the AI4HEALTHSEC. The system implements 5 main services:

- Manage Application Users (e.g., create, update, delete simple users, admins, etc.);
- Store samples' data;
- Store samples' logistics;
- Store data for sample storage; and
- Store user data (e.g., sample owners, related sample operators/researchers, etc.).

For the implementation of the main functionality, the core UBA-PVS server can run Windows or Linux OS and deploys Apache Tomcat v7 Webserver, PostgresSQL v9.2 or 9.3, pgAdmin Tool, and JRE7. The user logins the system via a web browser (i.e., IE 11 (or higher) and Firefox 32 (or higher)).

Moreover, 4 user roles are modelled. The *Standard User* only reads data (e.g., view data and export, execute queries, etc.). The *Sample Agent* writes on selected data sets (e.g., samples). The *Sample Manager* writes on sample-specific basis data (e.g., sample repository, sample kind, etc.) or deletes samples. The *Application Administrator* writes on the basis data and acts as a superuser of the system.

4.2 Demo deployment

In this demonstration, possible cyberattacks were executed to the piloting assets. The security events were monitored using the Event Captors, the information was distributed to

Fig. 6 Demo deployment

the related agents (PAs/SAs). Metadon aggregates the data, capable to transform the payloads and stores them to the Elasticsearch. That way the IDS alerts are possible to be parsed by the Assurance Platform and the other AI4HEALTHSEC components.

A depiction of the use-case scenario is presented in Fig. 6. Several cyberattacks are performed, including: the execution of a DoS attack using hping3, an UDP Port Scanning using Nmap, on-line password guessing, identity theft, encrypting files with ransomware, etc. All these actions were successfully detected and mitigated by the system.

The alerts generated by the intrusion detection mechanisms are retrieved and collected to the AI4HEALTHSEC as explained before. Then the events are parsed to the Assurance Platform and are visualized to the Main Dashboard. The IDS alerts are defined using the following rules. The rules can be configured according to the needs, and the intention of the integration was to test a basic ruleset and then extend to a larger rule dataset. Automated responses can be also defined, as well as the alerting of affected organizations in the swarm network and the sharing of CTI with collaborating communities.

4.3 Core reasoning with rule-based logic

As mentioned before, main automated detection and analysis processes is performed by the smart agents. Data gathering is accomplished via Beats and customized Event Captors. Then, the agents process these pieces of knowledge and reasons about the current status of the system. This includes the assessment of criteria or policies for security, privacy, or other properties, compliance with Service Level Agreements (SLAs), computation of metrics (e.g., service up-time, mean time to response (MTTResp), mean time to restore MTTRest, etc.), as well as attack assessment.

Indicative use case scenarios are described below. The description is in EC and their implementation in Drools. All these examples are assessed simultaneously and can be customized or combined in order to tackle more use cases if required. Also, there are general rules that digest alerts coming from Snort (or other intrusion detection and alerting mechanisms).

4.3.1 Confidentiality property criterion—the users access a service from a set of white-listed IPs

A Filebeat or Auditbeat captures the user interaction with a service (or other resource). If a user access is recorded from a different IP, it can be due to some attack that manage to overcome the deployed defenses (e.g., firewall) and infiltrate the system. The EC theory for this assessment is consisted of 2 rules:

- *Rule*1: if there is a call request of the service (_serviceName) at some timepoint (_t) from a user (_userName) who access the system from an IP (_IP), and this IP is also denoted in the whitelist with the legitimate IPs (defined as a fluent), then record a success.

Happens(Event(_e, call (_serviceName, _userName, _IP), _t, [_t, _t]) ∧ HoldsAt(Fluent(_f, IPsWL (_serviceName, _IPsList), _t, [_t, _t]) ∧ Contains(_IPsList, _IP) =>Initiates(Event(_e), Fluent(SuccessfullUse), _t)

- *Rule*2: if there is a call request of the service (_serviceName) at some timepoint (_t) from a user (_userName) who access the system from an IP (_IP), and this IP has not been denoted in the whitelist with the legitimate IPs (defined as a fluent), then record a violation.

Happens(Event(_e, call (_serviceName, _userName, _IP), _t, [_t, _t]) ∧ HoldsAt(Fluent(_f, IPsWL (_serviceName, _IPsList), _t, [_t, _t]) ∧ ¬Contains(_IPsList, _IP) => Initiates(Event(_e), Fluent(ViolatedUse), _t)

4.3.2 Confidentiality property & privacy criterions—a system resource (e.g., file or service call) is accessed only by a list of authorized users

A Filebeat or Auditbeat captures the resource interaction with the system's users. If an unauthorized user accesses the resource, it can be due to some attack (e.g., privilege escalation) that manage to overcome the deployed authorization techniques. The EC theory for this assessment is consisted of 2 rules:

- *Rule*1: if there is an access to a resource (_resourceName) at some timepoint (_t) from a user (_userName) and this user has also the access privileges to do so (defined as a fluent), then record a success.

Happens(Event(_e, access (_resourceName, _userName), _t, [_t, _t]) ∧ HoldsAt(Fluent(_f, authorizedUsers (_resourceName, _usersList), _t, [_t, _t]) ∧ Contains(_usersList, _userName)
Initiates(Event(_e), Fluent(SuccessfullUse), _t)

- *Rule*2: if there is an access to a resource (_resourceName) at some timepoint (_t) from a user (_userName) and this user has not the access privileges to do so (defined as a fluent), then record a violation.

Happens(Event(_e, access (_resourceName, _userName), _t, [_t, _t]) ∧ HoldsAt(Fluent(_f, authorizedUsers (_resourceName, _usersList), _t, [_t, _t]) ∧ ¬Contains(_usersList, _userName)
=> Initiates(Event(_e), Fluent(SuccessfullUse), _t)

4.3.3 Integrity property criterion—for every request on a specified service S2, there must have been called the service S1 first

A Filebeat reads the logfile of two monitored services S2 and S1, respectively. There is a criterion that the S1 must always be called before S2. If S2 has been called without the prior execution of S1, it can be due to an attack that managed to bypass the defined workflow or sequence (e.g., an SQL injection that requests data for users who are not logged in the system). The EC theory for this assessment is consisted of 2 rules:

- *Rule*1: if there is a call request of the service S2 at some timepoint (_t2) and there is also a relevant call on S1 (this is checked via the other event arguments which are the same for both events) at a past timepoint (_t1 + SLA_Threshold), then record a success.

Happens(Event(_e2, call(_S2, _opInst, _arg1, _arg2), _t2, [_t2, _t2]) ∧ Happens(Event(_e1, call(_S1, _opInst, _arg1, _arg3), _t1, (_0, _t2]
=> Initiates(Event(_e2), Fluent(SuccessfullCall), _t2)

- *Rule*2: if there is a call request of the service (_serviceName) at some timepoint (_t1) and there is not a relevant response (this is checked via the other event arguments which are the same for both events) within the acceptable time window (_t1 + SLA_Threshold), then record a violation.

Happens(Event(_e2, call(_S2, _opInst, _arg1, _arg2), _t2, [_t2, _t2]) ∧ ¬Happens(Event(_e1, call(_S1, _opInst, _arg1, _arg3), _t1, (_0, _t2]
=> Initiates(Event(_e2), Fluent(ViolationCall), _t2)

4.3.4 Integrity property criterion—there is only one active login session for each user on a service

A Filebeat reads the logfile of a monitored service. There is a criterion that each user can have only one active login. If a user has more than one active sessions, this could be due to an attacker that has gain access to the system and is currently on-line. The sessions must be further checked by the system operator. The EC theory for this assessment is consisted of 2 rules:

- *Rule*1: if there is a new login in the service (_serviceName) for a specific user (_userName), there must have been recorded a logout event for every previous successful login.

Happens(Event(_e1, login (_serviceName, _userName, _session1), _t1, [_t1, _t1]) ∧ Happens(Event(_e2, login (_serviceName, _userName, _session2), _t2, (_t1, _t2]) ∧ Happens(Event(_e3, logout (_serviceName, _userName, _session1), _t3, (_t1, _t2))
=> Initiates(Event(_e2), Fluent(SuccessfullCall), _t2)

- *Rule*2: if there is a new login in the service (_serviceName) for a specific user (_userName) and there is a past logged in session that has not been ended yet, then record a violation.

Happens(Event(_e1, login (_serviceName, _userName, _session1), _t1, [_t1, _t1]) ∧ Happens(Event(_e2, login (_serviceName, _userName, _session2), _t2, (_t1, _t2]) ∧ ¬Happens(Event(_e3, logout (_serviceName, _userName, _session1), _t3, (_t1, _t2))
=> Initiates(Event(_e2), Fluent(ViolationCall), _t1)

4.3.5 Availability property SLA—For every request on a specified service, there is a response within a specified time window

A Filebeat reads the logfile of a monitored service. There is a Service Level Agreement (SLA) that the service must respond each request within a specified period. Failure to deliver the service on time could be to congestion, system failure or breakdown, and/or malicious disruption (e.g., flooding attacks). The EC theory for this assessment is consisted of 2 rules:

- *Rule*1: if there is a call request of the service (_serviceName) at some timepoint (_t1) and there is also a relevant response (this is checked via the other event arguments which are the same for both events) within the acceptable time window (_t1 + SLA_Threshold), then record a success.

Happens(Event(_e1, call (_serviceName, _serviceInst, _arg1, _arg2), _t1, [_t1, _t1]) ∧ Happens(Event(_e2, res (_serviceName, _ serviceInst, _arg1 , _arg2), _t2, [_t1, _t1+SLA_Threshold])
=> Initiates(Event(_e2), Fluent(SuccessfullResponse), _t2)

- *Rule*2: if there is a call request of the service (_serviceName) at some timepoint (_t1) and there is not a relevant response (this is checked via the other event arguments which are the same for both events) within the acceptable time window (_t1 + SLA_Threshold), then record a violation.

Happens(Event(_e1, call (_serviceName, _serviceInst, _arg1, _arg2), _t1, [_t1, _t1]) ∧ ¬Happens(Event(_e2, res (_serviceName, _ serviceInst, _arg1 , _arg2), _t2, [_t1, _t1+SLA_Threshold])
=> Initiates(Event(_e2), Fluent(ViolatedResponse), _t2)

4.3.6 Availability property SLA—A service must be available and must not be down for more than a pre-defined threshold

A Heartbeat or customized Event Captor (i.e., get the HTTP status) that checks the status of a service. There is a Service Level Agreement (SLA) that the service must be up and running, and in case of unavailability, the service administrator/operator has a maximum pre-defined time window (e.g., 1 h) to fix the problem and restore the proper operation. Service unavailability could be to congestion, system failure or breakdown, and/or malicious disruption (e.g., Denial of Service (DoS) attack). The EC theory for this assessment is consisted of 4 rules:

- *Rule*1: if the status check for a service (_serviceName) at some timepoint (_t1) is normal, then record a success.

Happens(Event(_e, serviceStatus (_serviceName, "Available"), _t, [_t, _t])
=> Initiates(Event(_e), Fluent(AvailableService, _serviceName), _t)

- *Rule*2: if the status check for a service (_serviceName) at some timepoint (_t1) is unavailable, then record a violation and start checking against the SLA threshold (Rules 3 and 4).

Happens(Event(_e, serviceStatus (_serviceName, "Unavailable"), _t, [_t, _t]) ∧ ¬HoldsAt(Fluent(_f, UnavailableService (_serviceName), _t, [_t, _t])
=> Initiates(Event(_e), Fluent(UnavailableService, _serviceName), _t)

- *Rule*3: if the service (_serviceName) was unavailable and the operation was restored within the pre-defined time window (_PreDefinedThreshold) of the SLA, then record a success.

Happens(Event(_e, serviceStatus (_serviceName, "Available"), _t2, [_t2, _t2]) ∧ HoldsAt(Fluent(_f, UnavailableService (_serviceName), _t1, [_t1, _t2]) ∧ eval(_t2-_t1<=_PreDefinedThreshold)
=> Terminates(Event(_e), Fluent(_f), _t2) ∧ Initiates(Event(_e), Fluent(SuccessfulServiceRestore, _serviceName), _t2)

- *Rule*4: if the service (_serviceName) was unavailable and the operation was not restored within the pre-defined time window (_PreDefinedThreshold) of the SLA, then record a violation.

Happens(Event(_e, serviceStatus (_serviceName, "Available"), _t2, [_t2, _t2]) ∧ HoldsAt(Fluent(_f, UnavailableService (_serviceName), _t1, [_t1, _t2]) ∧ eval(_t2-_t1>_PreDefinedThreshold)
=> Terminates(Event(_e), Fluent(_f), _t2) ∧ Initiates(Event(_e), Fluent(ViolatedServiceRestore, _serviceName), _t2)

4.3.7 Integrity and Availability property—Observe potential ransomware activity on system resources

When a ransomware is activated, it will start to recursively read and encrypt high volumes of data. A customized Event Captor periodically observes (e.g., 1 min) the volume of files in a folder that are accessed within the current time window. If this volume goes beyond a pre-defined threshold (e.g., 100 accesses), notify for a potential ransomware activity. The EC theory for this assessment is consisted of 1 rule:

- *Rule*1: if the file access volume check for a folder with valuable data (_folderName) at some timepoint (_t) is beyond a pre-define threshold (_PreDefinedThreshold), then record a violation.

Happens(Event(_e, accessVolume (_folderName, _measuredVolume), _t, [_t, _t]) ∧ eval(_measuredVolume >_PreDefinedThreshold)
=>Initiates(Event(_e), Fluent(SuspiciousRansomwareActions, _folderName), _t)

4.3.8 Metrics—Compute usage metric

The previous rules can be further extended (mostly the Availability criterions) in order to estimate measurable variables for system or service usage. Indicative examples include: (1) the total up-time for a monitored period, (2) the mean time to respond, and (3) the mean time to restore. The EC theory for these assessments is consisted of 6 rules, respectively.

4.3.8.1 Total up-time for a service

- *Rule*1: if there is a new monitoring period (e.g., every year or month), then initiate the total up-time (maintained as the fluent totalUpTime) for the service (_serviceName).

Happens(Event(_e, newPeriod (_serviceName), _t, [_t, _t]) ∧ HoldsAt(Fluent(_f, totalUpTime, _serviceName, _value), _t)
=> Terminates(Event(_e), Fluent(_f), _t) ∧ Initiates(Event(_e), Fluent(totalUpTime, _serviceName, 0), _t)

- *Rule*2: if an examined service (_serviceName) is up and running at some timepoint (_t) within the current monitoring period (e.g., running year), then update the total up-time metric accordingly (maintained as the fluent totalUpTime).

Happens(Event(_e, serviceStatus (_serviceName, "Available"), _t2, [_t2, _t2]) ∧ HoldsAt(Fluent(_f, totalUpTime, _serviceName, _value), _t1)
=>Terminates(Event(_e), Fluent(_f), _t) ∧ Initiates(Event(_e), Fluent(totalUpTime, _serviceName, (_t2—_t1 + _value), _t)

4.3.8.2 Mean time to respond (MTTResp)

- *Rule*1: if there is a new monitoring period (e.g., every year or month), then initiate the MTTResp (maintained as the fluent MTTResp) for the service (_serviceName).

Happens(Event(_e, newPeriod (_serviceName), _t, [_t, _t]) ∧ HoldsAt(Fluent(_f, MTTResp, _serviceName, _value), _t)
=>Terminates(Event(_e), Fluent(_f), _t) ∧ Initiates(Event(_e), Fluent(MTTRep, _serviceName, func_InitMTTResp(_serviceName)), _t)

(func_InitMTTResp is a method in Java that initializes an object that maintains an internal data structure for the response times of the service (_serviceName) and calculates the MTTResp. Initially, it is 0).

- *Rule*2: if there is a new response (_res) on a previous call (_call) for a service (_serviceName), then update the MTTResp accordingly (maintained as the fluent MTTResp).

Happens(Event(_e2, res (_serviceName, sessionID), _t2, [_t2, _t2]) ∧ Happens(Event(_e1, call (_serviceName, sessionID), _t1, [_t1, _t2)) ∧ HoldsAt(Fluent(_f, MTTR, _serviceName, _value), _t1)
=> Terminates(Event(_e2), Fluent(_f), _t2) ∧ Initiates(Event(_e2), Fluent(MTTResp, _serviceName, (func_UpdateMTTResp(_serviceName, _t2—_t1)), _t2)

(func_UpdateMTTResp is a method in Java for the object that maintains the response times of the service (_serviceName) in an internal data structure, adds the new response time (_t2—_t1), and calculates the MTTResp).

4.3.8.3 Mean time to restore (MTTRest)—in combination with availability criteria of the subsection 4.3.6)

- *Rule*1: if there is a new unavailability period for a service (_serviceName), then initiate the MTTRest (maintained as the fluent MTTRest) metric.

Happens(Event(_e, serviceStatus (_serviceName, "Unavailable"), _t, [_t, _t]) ∧ ¬HoldsAt(Fluent(_f1, UnavailableService (_serviceName), _t, [_t, _t]) ∧ HoldsAt(Fluent(_f2, MTTRest, _serviceName, _value), _t)
=> Terminates(Event(_e), Fluent(_f2), _t) ∧ Initiates(Event(_e), Fluent(MTTRest, _serviceName, func_InitMTTRest(_serviceName,_t)), _t)

(func_InitMTTRest is a method in Java that initializes an object that maintains an internal data structure for the unavailable periods of the service (_serviceName). The metric starts to count when the unavailable period is observed (_t) and will calculate the MTTRest when the service is later restored).

- *Rule*2: if a previously unavailable service (_serviceName) is now restored, then update the MTTResp accordingly (maintained as the fluent MTTResp).

Happens(Event(_e, serviceStatus (_serviceName, "Available"), _t2, [_t2, _t2]) ∧ HoldsAt(Fluent(_f, UnavailableService (_serviceName), _t1, [_t1, _t2]) ∧ HoldsAt(Fluent(_f, MTTR, _serviceName, _value), _t1)
=> Terminates(Event(_e2), Fluent(_f), _t2) ∧ Initiates(Event(_e2), Fluent(MTTResp, _serviceName, (func_UpdateMTTRest(_serviceName, _t2—_t1)), _t2)

(func_UpdateMTTRest is a method in Java for the object that maintains the unavailable times of the service (_serviceName) in an internal data structure, adds the new restore time (_t2—_t1), and calculates the MTTRest).

4.4 Enhance reasoning with ML

Attacks prediction concerns the identification of possible scenarios of future attacks through forecasting models. As it has already mentioned, this sub phase includes reasoning on the fused data and identification of on-going and future attacks. During this phase, the fused data about the situation of the system are evaluated by comparing them with models of normal operation and attack scenarios.

An analysis of threats and vulnerabilities was carried out using ML algorithms [35], including the BERT neural language model and XGBoost. These models were utilized to gather current information from Natural Language documents widely accessible online, while simultaneously assessing the severity of the detected threats and vulnerabilities affecting the healthcare system. The practical application of this method involved analyzing cybersecurity news from the Hacker News website and examining reports on CVEs.

Concerning network and computer security, open datasets were utilized from [97], covering datasets for intrusion detection from 1999 to 2020, like the ones from WUSTL EHMS 2020 [98] and DARPA [99]. These datasets contain traces from IP, TCP, UPD, and ICMP protocols, including several types of attack, like Denial of Service (DoS), User to Root (U2R), Remote to Local (R2L), and Probe or Scan. Therefore, ML models were trained based on Neural Networks (Multilayer Perceptron Classifier), which could be utilized for the evaluation of network traffic. Moreover, models from the normal operation of the AI4HEALTHSEC piloting systems were used, including User and Entity Behavior Analytics (UEBA).

Thereupon, the analysis and decision-making aspects of the AI4HEALTHSEC can be further enhanced with system and security analytics. The Agents can leverage its ML elements in order to support fault and attack prediction functionality.

4.5 System faults forecasting

For example, once there are adequate ML data (i.e., for training and evaluation), the ML and AutoML modules could forecast service or component breakdowns and upcoming unavailability events based on the historic information that has been recorded for the specific service or component. The prediction of such incidents could further fortify the abovementioned incident assessment procedures and trigger proactive actions to reduce their occurrence.

4.6 Attack forecasting based on IOA

Similarly, the agents could evaluate the ongoing system operation and identify Indicators Of Attack (IOA), prior the Indicators Of Compromised (IOC). For instance, an attacker may be performing an on-line guessing attack for a username on one of the monitored services. This could be captured in the form of several subsequent failed login attempts (e.g., admin/admin, admin/12345678, etc.). The agent will raise a warning for the specific username (i.e., admin). If the attack is successful, eventually the attacker will discover a valid username/password pair and there will be a successful login attempt. Without analytics functionality, we would have reasoned that this final event constitutes a legitimate login, as the used credentials are the correct ones. However, with the on-line guessing attack raised for the username admin during the current time-window, the system successfully identifies that this is an attack. Mitigation actions could be performed afterwards (e.g., block the username account and the attacker, inform the platform operator and the user for the compromised credentials, etc.).

4.7 Attack forecasting based on Anomaly detection and UEBA

The integrated Assurance Platform with the rule-based monitoring and the ML enhancements can be also utilized for anomaly detection activities. The modern User and Entity Behavior Analytics (UEBA) is such a case. UEBA models are trained in order to detect the activities of a user or other system entity. If the runtime behavioral patterns do not comply with the previous behavior, an anomaly is detected. As an indicative scenario for AI4HEALTHSEC, it is considered the protection SSH login service of the users to platform or piloting healthcare services. Usually, the employees of an organization or the users of a service (e.g., email) access the service from specific devices, locations, and/or working hours. For example, the web services of public hospital in Heraklion in Greece should be accessed by IP locations in Heraklion. Based on his/her routine, if a user or employee logins a service from another region or country (i.e., Maldives, China), this could be a suspicious event. Popular sites (Google, Facebook, LinkedIn, etc.) detect such events for their services. At initialization, the ML parses the organization's logfiles and discloses the usual IPs, devices (e.g., based on the MAC details), working hours, and other pieces of information that are utilized by the current users, as well as other UEBA-related information. At runtime, the Monitor will examine every successful service login and request the AutoML module to check if the login action for the specific user is complying with the related UEBA profile. The EC theory for these assessments is consisted of 3 rules.

- *Rule*1: if there is a new login to the service (_serviceName), then inform the AutoML module to assess the event. This is performed by the Executive Event 'Apply_ML', which will examine in the internal database if there is a ready

ML model for this user (_userName) and service (_serviceName), and if yes, then it will send a relevant event/request to the AutoML.

Happens(Event(_e1, login (_serviceName, _userName, _loginTime, _IP, …), _t, [_t, _t])
=>HappensExec(Event(_e2, Apply_ML (_serviceName, _userName, _loginTime, _IP, …), _t, [_t, _t]))

- *Rule*2: If there is a service login for a username (_userName) and a ML check login confirmation with high confidence (probability > = 0.9), then record a successful login.

Happens(Event(_e1, login (_serviceName, _userName, _loginTime, _IP, …), _t, [_t, _t])) ∧ Happens(Event(_e2, MLCheckLogin (_serviceName, _probability), _t2, [_t2, _t2]) ∧ eval(_probability > = 0.9)
=>Initiates(Event(_e2), Fluent(SuccessfulLogin, _serviceName, _userName), _t)

- *Rule*3: If there is a service login for a username (_userName) and a ML check login confirmation with low confidence (probability < 0.9), then record a suspicious login.

Happens(Event(_e1, login (_serviceName, _userName, _loginTime, _IP, …), _t, [_t, _t])) ∧ Happens(Event(_e2, MLCheckLogin (_serviceName, _probability), _t2, [_t2, _t2]) ∧ eval(_probability < 0.9)
=> Initiates(Event(_e2), Fluent(ViolatedLogin, _serviceName, _userName), _t)

4.8 Incident handling actions

Towards the first prototype of the incident handling process, Metadon and the Assurance Platform can automatically perform some pre-defined response strategies to specific problematic or malicious cases. As sketched in the introductory sections, this comes in the form of parameterized scripts for Metadon and routines of JAVA code for the Assurance Platform. Moreover, the Assurance Platform, when performing part of the internal swarm intelligence, can command the local Metadon instances and trigger some of these actions. In a semi-automated fashion, this process can be also activated by the end user of the AI4HEALTHSEC platform via the provided GUI. Eventually, the user can combine information from the incident handling and risk assessment processes and perform manually complex containment, eradication, and recovery procedures that are not covered by the automated/semi-automated modules.

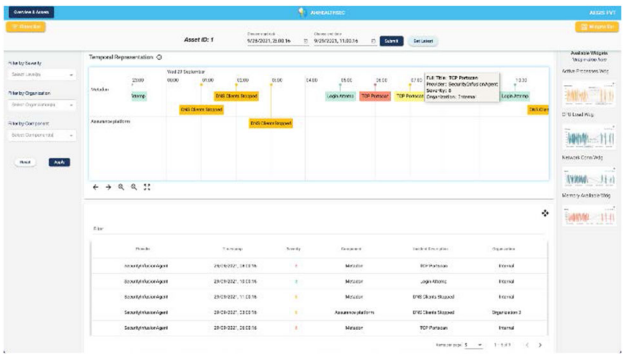

Fig. 7 FVT and Unified Dashboard

Indicative response actions that are considered for the first implementation of the incident handling process, include:

- Metadon (PAs):
  - o Script to change firewall rules and mitigate Denial of Service (DoS) attacks or isolate compromised equipment.
  - o Send email to local system user/operator when a specified malicious or suspicious action is observed.
- Assurance Platform (SAs):
  - o Send email/message to platform user/operator when a specified malicious or suspicious action is observed.
  - o Inform collaborating organizations of the swarm network.
  - o Gather information for identified security incidents and prepare a draft CTI message.
- End user:
  - o Utilize FVT to perform a digital forensics analysis for identified security incidents.
  - o Finalize the CTI elements and authorize their distribution to the rest community (e.g., via the MISP service).
  - o Perform forward or backward recovery actions on main computerized assets.

The Fig. 7 depicts the output of FVT and the Unified Dashboard, where the user can review all these details.

4.9 CTI Reasoning

The MISP AI4HEALTHSEC module, a MISP instance, provides functionalities related to cyber-security information and deploys the connection between AI4HEALTHSEC with the central MISP instance managed by the CONCORDIA project, which is the largest major European cyber-security consortium and has a mission to establish an EU-integrated cyber-security ecosystem for digital sovereignty in Europe.

The EC theory for the filtering of MISP-related events is consisted of 3 rules. Rules 1 and 2 maintain a list of MISP-related events, by adding and removing event types in the list, respectively. Rule 3 is performing the actual filtering functionality for each occurred event and the call of the MISP Contributor internal API to share the information.

- *Rule*1: if there is a new request (add_to_MISP_events_list) to include an event type (_eventName) in the filtering mechanism, then update the list (_hashset) accordingly. The list is a hashset and will add the event type only once.

Happens(Event(_e, add_to_MISP_events_list(_eventName), _t, [_t, _t])) ∧ HoldsAt(Fluent(_f, MISP_events_list (_hashset))
=>_hashset.add(_eventName)

- *Rule*2: if there is a new request (remove_to_MISP_events_list) to remove an event type (_eventName) from the filtering mechanism, then update the list (_hashset) accordingly. The list is a hashset datatype and will remove the event type only if it has been previously added.

Happens(Event(_e, remove_to_MISP_events_list(_eventName), _t, [_t, _t])) ∧ HoldsAt(Fluent(_f, MISP_events_list (_hashset))
=> _hashset.remove(_eventName)

- *Rule*3: If there is recorded a new event (_e) and its event type (_e.name) is included in the MISP-related events (_f._hashset), then call the MISP Contributorclient internal API to share the related information.

Happens(Event(_e), _t, [_t, _t])) ∧ HoldsAt(Fluent(_f, MISP_events_list (_hashset)) ∧ eval(_hashset.contains(_e.name))
=> {Call MISP client internal API to send _e}

When relevant events are recorded by the agents, the event is disseminated automatically to the AI4HEALTHSEC MISP module. The events can be also disseminated automatically to the CONCORDIA MISP, but in general it is preferred that the analyst will review the overall information, form an incident report, and define the sharing approach.

The Fig. 8 depicts the dissemination of information from AI4HEALTHSEC to the CONCORDIA community.

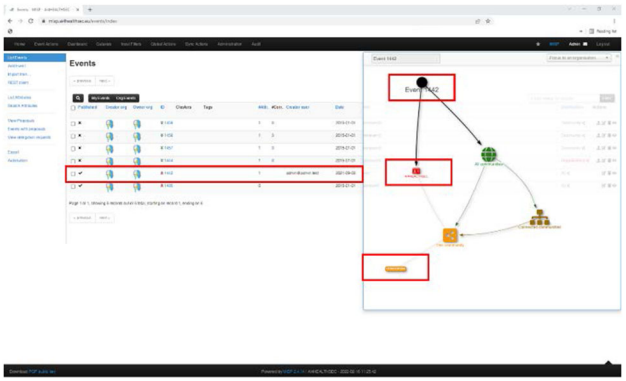

Fig. 8 Sharing Events between MISP AI4HEALTHSEC and MISP CONCORDIA

5 Discussion and future directions

Sharing Cyber Threat Intelligence (CTI) has emerged as a promising approach to raise situational awareness among stakeholders and tackle cyber-threats proactively rather than reactively [100]-[101]. Organizations may be required to adopt a threat intelligence program and share information to survive current and future attacks. Failure to share information about known threats may result in stakeholders being held responsible for breaches caused by those threats. The primary objective of sharing threat intelligence is to promote situational awareness among stakeholders, enabling them to swiftly implement remedies for emerging threats and vulnerabilities. CTI can also help stakeholders in making tactical decisions. However, implementing a CTI program that can consume and disseminate information in a timely manner is a significant challenge for practitioners. Furthermore, stakeholders face difficulties in making CTI relevant to their system. One of the most significant challenges faced by practitioners before sharing their CTI is how to comprehend the information and apply its remedy. Stakeholders prefer an automated and effective sharing process, but lack of models and tools makes it difficult [102]. However, sharing information about vulnerabilities through manual means, such as stakeholder-to-stakeholder sharing or through trusted groups like an ISAC, is still a common approach to create situation awareness among stakeholders and quickly alert them to threats. Yet, manual sharing can be ineffective due to slow sharing, errors during processing, or subjective filtering. Automating some processes could enhance the effectiveness of CTI sharing. CTI sharing occurs on a global scale, but different laws and regulations across countries create challenges. The survey in [29] identifies current challenges that impede the sharing process, such as trust, reputation, relevance, anonymity, timeliness, and data interoperability. Before critical threat intelligence is shared, trusted relationships must be established. Governance, management, policies, and legal factors may also impact CTI sharing. Though most sharing occurs on a national level, international exchanges are gaining momentum. Challenges regarding

human behavior, cultural and language barriers, and incentives are also discussed. Organizations need to invest time and resources into understanding the importance of CTI sharing programs and building them for the future [29].

More and more, cyber-attacks are causing harm to businesses by exploiting vulnerabilities in networked manufacturing machines. In certain instances, these attacks on vital industrial equipment have the ability to compromise the overall business model. It is a competitive advantage to identify and assess in advance the primary assets that are at risk of cyber-attacks and the potential business consequences [2].

The private and public sectors are increasingly interested in using AI to address cyber-security challenges. Market projections suggest that the AI cyber-security market will experience tremendous growth, rising from $1 billion in 2016 to an estimated $34.8 billion by 2025. AI is specifically mentioned in the latest national cyber-security and defense strategies of various governments. Efforts to establish new standards and certification procedures to build trust in AI are also underway worldwide. However, trusting AI in security/privacy is a double-edged sword [102]. AI in cyber-security has its advantages and disadvantages. While it can enhance cyber-security, it can also expose AI applications to new forms of attacks (e.g., semantic attacks [103]), creating serious security threats. We suggest that having complete trust in AI for cyber-security is not justified, and to minimize security risks, it is essential to have some form of control to guarantee the use of 'reliable AI' for cyber-security. The study in [102] suggests three roadmaps in order to improve the AI applications for cyber-security and improve their robustness against wily manipulation: (1) promote in-house development as most common types of attacks are accommodated by the use of commercial solutions, (2) enhance the datasets with adversarial training, and (3) perform parallel and dynamic monitoring. AI4HEALTHSEC stipulates with these ideas and implements such functionality (i.e., in-house development, ML procedures and AI decision-making procedures that takes into account the potential infiltration by adversaries, and continuous monitoring of the involved system components).

Nonetheless, CTI cannot resolve all cyber security problems. The are many limitations of current CTI solutions (i.e., countering zero-day attacks [61–63]), as well as obstacles that need to be overcome (e.g., CTI incorporation rate, especially for organizations with low or very low cyber-security budget and culture that can be found in a supply-chain ecosystem, timely sharing of information for wily actions and trends). The AI4HEALTHSEC approach tries to tackle these issues. The overall incident handling operation can detect malicious campaigns from their early reconnaissance phase to their actual execution and exploitation functions. The organization is becoming aware of the danger and the human operator along with the automated analysis elements could potentially help in identifying new attacks and hacker tactics. The automated and semi-automated process for analysis and information sharing could enhance the robustness of supply-chains, even when small organizations with low security budget and expertise are engaged.

Moreover, modern approaches are highlighting the need to share malicious patterns and behavioral models, not just data or information of low value [104, 105]. This is also something that could be supported by our methodology in the modern Cyber Threat Intelligence for "Things" landscape [60, 106].

SHERPA is an EU funded project [107] which investigates the role of AI and Big Data analytics in Smart Information Systems (SIS) and how they are impacting ethics and human rights. They use of SIS in cyber-security is one of the examined domains. It is concluded that currently there is comparatively little work on this aspect, for reasons like the danger of false positives and negatives, the relatively low intelligence of existing systems, and the high diversity of attacks and malicious tactics.

Moreover, several ethical issues are raised from the adaptation and automation of cyber-security with AI, such as: the difficulty is supporting the proper informed consent procedures for users or other involved people, protection from harm, disclosure of vulnerabilities, biases, the nature of hacking, trust and transparency of the AI/ML algorithms, the necessity for a risk assessment in cybersecurity, responsibility between companies, government and users, lack of clear codes for international practice, as well as the issue of monetization (how far can one ethically go to monetize customer's data).

Despite these issues, there are valid reasons to consider employing SIS in cybersecurity. Their primary utility lies in examining systems for recognized attacks or unusual behavior patterns that strongly suggest a cyber-attack. When paired with a human operator who reviews any warnings to decide on the appropriate response, this integrated human—machine security system can be quite effective. However, it still encounters challenges related to automation bias and a high number of false alarms.

AI4HEALTHSEC is also concerning about those issues and have come in similar conclusions in terms of biased and false inference results of fully automated systems, lack of well-established methodologies for international cooperation, privacy concerns, as well as security and new opportunities and options for attackers on the AI counterparts themselves [87, 108, 109]. Therefore, the proposed solution for incident handling is mostly focused in identifying known vulnerabilities and abnormalities, while keeping the human operator in the security loop. Well-established methodologies for CTI modelling and sharing are supported in an attempt to enhance the effort of international collaboration and interoperable practices. Anonymization of data can be performed with the CHIMERA module, while the security

of the AI4HEALTHSEC Platform itself has also been taken into account.

Finally, AI4HELATHSEC adopted rule-based methods for the implementation of the core AI reasoning. Other alternatives can be examined as well, like policy-based languages (e.g., [110–112]). These may include programming or specification languages designed to define policies that govern the behavior of AI systems. Such languages are crucial in contexts where AI systems need to make decisions based on a set of predefined rules or policies, ensuring that the AI behaves in a manner that is consistent, predictable, and aligned with organizational or ethical guidelines. They are particularly important in areas such as access control, network management, autonomous systems, and any domain where AI systems interact with humans or make autonomous decisions. Therefore, incorporating such solutions in AI development helps in imposing a structured framework for AI behavior, ensuring that AI systems adhere to specified norms and regulations, and facilitating transparency and accountability in AI operations.

6 Conclusion

Cyber-security is one of the hot research topics at the time. Warfare and continuous attacks on global supply-chains and networks are making the protection of critical and other ICT infrastructures essential. This paper presents a swarm-intelligence approach for the automation of: (1) CTI incorporation, (2) dynamic risk assessment, (3) continuous monitoring, evaluation, and response to ongoing events, and iv) post-incident actions and feedback. Recent advancements of CTI technologies and models are marshaled (i.e., MISP, CVEs, STIX, etc.) with AI/ML automating the decision-making processes. The demonstration on healthcare settings reveals the effectiveness of the system in detecting and responding to attacks. Incorporating in a fast pace the latest distributed CTI, helps in advancing an organization's intelligence very quick and decreasing the time to detect and mitigate new (known) attacks. Moreover, the integrated ML solutions (e.g., UEBA models) can defend the system against some zero-day attacks, disclosing wily activity elements in a semi-automated fashion, which are immediately shared with CTI communities, limiting the attacker's effectiveness and benefits. Although, the solution has been deployed and tested in the healthcare domain, it is general and can be applied in other sectors as well.

Acknowledgements This work has received funding from the European Union's Horizon 2020 research and innovation programmes under grant agreements No. 883273 (AI4HEALTHSEC), No. 101021659 (SENTINEL), No. 957337 (MARVEL), and No. 101070599 (SecOPERA).

Author contributions G.H., E.L., M.A., S.P., S.K., A.A., D.A., and S.K. wrote the main manuscript. G.K., M.C., S.K, A.A., G.H. implement the solution and D.A. and S.K. set the piloting environment. All authors reviewed the manuscript. S.P., S.I., and G.S. supervise the research activities and review the document.

Funding Funding is detailed in the 'Acknowledgement' section below.

Data availability The datasets generated during and/or analyzed during the current study are not publicly available due to confidentiality terms of the funding projects Grand Agreements but are available from the corresponding author on reasonable request.

Declarations

Conflict of interest The authors have no competing interests to declare that are relevant to the content of this article.

Humans or Animals Research No research on humans or animals took place during this research. Therefore, no informed consent procedures were needed.

References

1. Nankervis, A., Connell, J., Montague, A., Burgess, J.: The Fourth Industrial Revolution. Springer, Singapore, pp. 1–239.
2. Corallo, A., Lazoi, M., Lezzi, M.: Cybersecurity in the context of Industry 4.o: A structured classification of critical assets and business impacts. Computer in Industry, Elsevier, **114**, 1–15 (2020)
3. Mukhopadhyay, I.: Cyber threats landscape overview under the new normal, ICT analysis and applications. pp. 729–736 Springer, (2022)
4. Ding, J. et al., Cyber threats to smart grids: review, taxonomy, potential solutions, and future directions. Energies, MDPI, **15**, 1–37.
5. Ramakrishna, K.: The global threat landscape in 2020. Counter Terrorist Trends Anal, RSIS **13**(1), 1–13 (2021)
6. Tounsi, W., Rais, H.: A survey on technical threat intelligence in the age of sophisticated cyber attacks. Comput Secur Elsevier **72**, 212–233 (2018)
7. Morrison, A.: Cyber security landscape 2022. Deloitte, February, pp. 1–15 (2022)
8. Sclette, D., Caselli, M., Pernul, G.: A comparative study on cyber threat intelligence: the security incident response perspective. IEEE Commun Surv Tutor IEEE **23**(4), 2525–2556 (2021)
9. Fortune Business Insights, Cyber security market size, share & COVID-19 impact analysis, fortune business insights 2022. Available on-line at: https://www.fortunebusinessinsights.com/industry-reports/cyber-security-market-101165 (Access on 16/10/2022).
10. Lella, I. et al., ENISA Threat Landscape 2023. ENISA, October, pp 1–161 (2023)
11. Ertan, A. et al., Cyber threats and NATO 2030: horizon scanning and analysis. NATO CCDCOE Publications, pp 1–267 (2020)
12. Singleton, C. et al., X-force threat intelligence index 2022. IBM, February, pp 1–59 (2022)
13. Raj Samani, et al., McAfee Labs Threat Report 04.21. McAfee Corporation, April, 2021, pp 1–24.
14. ESET, Cybersecurity trends 2021: Staying secure in uncertain times. ESET, March, pp 1–19 (2021)
15. Sharwood, S.: US Doj reveals Russian supply chain attack targeting energy sector. The Register, March, (2022)

16. Wang, P., Johnson, C.: Cybersecurity incident handling: a case Study of the Equifax data breach. Issues Inform Syst IACIS **19**(3), 150–159 (2018)
17. Shafqat, N., Masood, A.: Comparative analysis of various national cyber security strategies. Int J Comput Sci Inform Secur **14**(1), 129–136 (2016)
18. Carr, M.: Public-private partnerships in national cyber-security strategies. Int Affairs Wily **92**(1), 43–62 (2016)
19. A. Unwala, S. Ghori, "Brandishing the Cybered Bear: Informaiton war and the Russia-Ukraine conflict," Military Cyber Affairs, vol. 1, issue 1, article 7, 2015, pp. 1–11.
20. Willett, M.: The cyber dimension of the Russia-Ukraine war. Global Politics Strateg, Taylor, Francis **64**(5), 7–26 (2022)
21. Stitilis, D., Pakutinskas, P., Malinauskaite, I.: EU and NATO cybersecurity strategies and national cyber security strategies: a comparative analysis. Secur J Springer **30**, 1151–1168 (2017)
22. Eggers, S.: A novel approach for analyzing the nuclear supply chain cyber-attack surface. Nuclear Eng Technol Elsevier **53**(3), 879–887 (2021)
23. Urciuoli, L., Mohanty, S., Hintsa, J., Bockesteijn, E.G.: The resilience of energy supply chains: a multiple case study approach on oil and gas supply chain to Europe. Supply Chain Manage: An Int J **19**(1), 46–63 (2014)
24. Ramsdale, A., Shiaeles, S., Kolokotronis, N.: A comparative analysis of cyber-threat intelligence sources, formats and languages. Electron, MDPI **9**, 1–22 (2020)
25. Schlette, D., et al.: Measuring and visualizing cyber threat intelligence quality. Int. J. Inf. Secur. **20**, 21–38 (2021)
26. Mavroeidis, V., Bromander, S.: Cyber threat intelligence model: an evaluation of taxonomies, sharing standards, and ontologies within cyber threat intelligence. Eur Intell Secur Inform Conf (EISIC) (2017). https://doi.org/10.1109/EISIC.2017.20
27. Bahrami, P.N., et al.: Cyber kill chain-based taxonomy of advanced persistent threat actors: analogy of tactics, techniques, and procedures. J Inform Process Syst KIPS **15**(4), 865–889 (2019)
28. Dargahi, T., et al.: A cyber-kill-chain based taxonomy of crypto-ransomware features. J Comput Virol Hacking Tech Springer **15**, 277–305 (2019)
29. Wagner, T.D., Mahbub, K., Palomar, E., Abdallah, A.E.: Cyber threat intelligence sharing: survey and research directions. Comput Secur Elsevier **87**, 1–27 (2019)
30. Barnum, S. (2014) Structured threat information expression (STIXTM). MITRE Corporation 1–22
31. Yeng, P.K., et al.: Data-driven and artificial intelligence (AI) approach for modelling and analyzing healthcare security practice: a systematic review. Intell Syst Appl Springer AISC **1250**, 1–18 (2020)
32. Yeng, P. K. et al., (2019) Framework for healthcare security practice analysis, modeling and incentivization. Int Conf on Big Data (Big Data) IEEE 3242–3251
33. Health-ISAC, Collaborating for resilience in healthcare—annual report 2022. Health-ISAC, 2022, pp. 1–28. Available on-line at: https://h-isac.org/wp-content/uploads/2023/04/2022_Health-ISAC-Annual-Report-sm.pdf (Access on 23/10/2023).
34. Basheer, R., Alkhatib, B.: Threats from the dark: a review over dark web investigation research for cyber threat intelligence. J Comput Networks Commun Hindawi **2021**, 1–21 (2021)
35. Silvestri, S., et al.: A machine learning approach for the NLP-based analysis of cyber threats and vulnerabilities of the healthcare ecosystem. Sensors MDPI **23**(2), 1–26 (2023)
36. Silvestri, S., et al.: Cyber threat assessment and management for securing healthcare ecosystems using natural language processing. Int J Inform Secur Springer **23**, 31–50 (2024)
37. Ponemon Institute LLC, Cyber security in operational technology: 7 insights you need to know, march 2019. Ponemon Institute LLC, (2019)
38. Taddeo, M.: Is cybersecurity a public good? Mind Mach Springer **29**, 349–354 (2019)
39. ISO/IEC (2016). ISO/IEC 27035–1:2016. Available on-line at: https://www.iso.org/standard/60803.html (Access on 23/10/2023).
40. ISO/IEC (2016). ISO/IEC 27035–2:2016. Available on-line at: https://www.iso.org/standard/62071.html (Access on 23/10/2023).
41. Barrett, M. P.: Framework for improving critical infrastructure cyber security. National Institute of Standards and Technology, Gaithersburg, Version 1.1, MD, USA (2018)
42. Scarfone, K., Grance, T., Masone, K.: Computer security incident handling guide. NIST Spec. Publ. **800**(61), 38 (2008)
43. West-Brown, M. J., Stikvoort, D., Kossakowski, K. P., Killcrece, G., Ruefle, R.: Handbook for computer security incident response teams (CSIRTs). Carnegie-Mellon Univ Pittsburgh Pa Software Engineering Inst (2003)
44. West-Brown, M., Stikvoort, D., Kossakowski, K., Killcrece, G., Ruefle, R.: Handbook for computer security incident response teams (csirts). DTIC Document, Tech. Rep., (2003)
45. Alberts, C., Dorofee, A., Killcrece, G., Ruefle, R. Zajicek, M.: Defining incident management processes for csirts: a work in progress. (2004)
46. Hashemi, Sayed Hadi, et al.: A comprehensive semi-automated incident handling workflow. 6th International Symposium on Telecommunications (IST). IEEE, (2012)
47. ENISA (2010) The European union agency for cybersecurity (ENISA) have provided a good practice guide for incident management. Available on-line at: https://www.enisa.europa.eu/publications/good-practice-guide-for-incident-management (Access on 23/10/2023).
48. Network, Europe. "Information security agency." Good practice guide for incident management 110 (2010)
49. Sadoddin, R., Ghorbani, A.: Alert correlation survey: framework and techniques. In Proceedings of the 2006 International Conference on Privacy, Security and Trust: Bridge the Gap Between PST Technologies and Business Services. ACM, 2006, pp. 37.
50. ISO/IEC 27039 (2015). Information technology—Security techniques—Selection, deployment, and operations of intrusion detection systems (IDPS). Available on-line at: https://www.iso.org/standard/56889.html (Access on 23/10/2023).
51. ISO/IEC 27041 (2015) Information technology—Security techniques—Guidance on assuring suitability and adequacy of incident investigative method. Available on-line at: https://www.iso.org/standard/44405.html (Access on 23/10/2023).
52. ISO/IEC 27042 (2015). Information technology—Security techniques—Guidelines for the analysis and interpretation of digital evidence. Available on-line at: https://www.iso.org/standard/44406.html (Access on 23/10/2023).
53. CRR Supplemental resource guide (2016). Volume 5 incident management Version 1.1, Carnegie Mellon University. Available on-line at: https://www.cisa.gov/publication/crrsupplemental-resource-guides (Access on 23/10/2023).
54. ITU-T X.1216 TELECOMMUNICATION STANDARDIZATION SECTOR OF ITU (09/2020) SERIES X (2020). DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY Cyberspace security—Cybersecurity Requirements for collection and preservation of cybersecurity incident evidence. Available on-line at: https://www.itu.int/rec/T-REC-X.1216-202009-I/en (Access on 23/10/2023).
55. Northcutt, S. Institute, S.: Computer security incident handling: step by step, a survival guide for computer security incident handling. Sans Institute, (2001)

56. Common Vulnerabilities and Exposures (CVE), MITRE, 2023. Available on-line at: cve.mitre.org (Access on 23/10/2023)
57. Tundis, A. Ruppert, S. Muhlhauser, M.: On the automated assessment of open-source cyber threat intelligence sources. International Conference on Computational science (ICC), Computational science—ICCS 2020, Springer, LNTCS, vol. 12138, 2020, pp. 453–467.
58. Papastergiou, S., Polemi, N.: MITIGATE: a dynamic supply chain cyber risk assessment methodology. Smart Trends Syst, Secur Sustain Springer, LNNS **18**, 1–9 (2017)
59. Schauer, S., Polemi, N., Mouratidis, H.: MITIGATE: A dynamic supply chain cyber risk assessment methodology. J Transport Secur, Springer **12**, 1–35 (2019)
60. Wagner, T. D. Cyber threat intelligence for "Things". International conference on cyber situational awareness, data analytics and assessment (Cyber SA), IEEE, Oxford, UK, (2019) pp. 1–6
61. Kumar, V., Sinha, D.: A robust intelligent zero-day cyber-attack detection technique. Complex Intell Syst Springer **7**, 2211–2234 (2021)
62. Zoppi, T., Ceccarelli, A., Bondavalli, A.: Unsupervised algorithms to detect zero-day attacks: strategy and application. IEEE Access IEEE **9**, 90603–90615 (2021)
63. Duessel, P., et al.: Detecting zero-day attacks using context-aware anomaly detection at the application-layer. Int J Inform Secur Springer **16**, 475–490 (2017)
64. ISO-31000:2018: Risk management, ISO, 2018. Available on-line at: www.iso.org/iso-31000-risk-management.html (accessed on 23/10/2023).
65. ISO-27001:2022: Information security management system, ISO/IEC, 2022. Available on-line at: www.iso.org/standard/27001 (accessed on 23/10/2023).
66. CIS Critical security controls, CIS. Available on-line at: https://www.cisecurity.org/controls (accessed on 23/10/2023).
67. Common vulnerability scoring system (CVSS) v4.0, FIRST, 2023. Available on-line at: https://www.first.org/cvss/v4-0/ (accessed on 23/10/2023).
68. Common platform enumeration (CPE), NIST, 2023. Available on-line at: nvd.nist.gov/products/cpe (accessed on 23/10/2023).
69. Common attack pattern enumeration and classification (CAPEC), MITRE, 2019. Available on-line at: capec.mitre.org (accessed on 23/10/2023).
70. Coordinated vulnerability disclosure (CVD), UK national cyber security centre (NCSC), 2018. Available on-line at: https://www.enisa.europa.eu/news/member-states/WEB_115207_Brochure NCSC_EN_A4.pdf (accessed on 23/10/2023).
71. Islam, S., Papastergiou, S., Kalogeraki, E.-M., Kioskli, K.: Cyber-attack path generation and prioritisation for securing healthcare systems. Appl Sci MDPI **12**, 1–22 (2022)
72. Hatzivasilis, G. et al., Continuous security assurance of modern supply-chain ecosystems with application in autonomous driving. IEEE CSR Workshop on Cyber Resilience and Economics (CRE), IEEE, Venice, Italy, 31 July—2 August (2023), pp. 1–6
73. CyberSANE, D2.1: Cyber Incident handling Trend Analysis. pp. 1–76, (2020)
74. E.T. Muller, Commonsense reasoning: an event calculus based approach. 2nd edn. M. Kaufmann, (2015)
75. Drools reasoning engine. Available on-line at: https://drools.org/ (accessed on 23/10/2023)
76. AutoKeras. Available on-line at: https://autokeras.com/ (accessed on 23/10/2023)
77. ELK Stack. Available on-line at: https://www.elastic.co/what-is/elk-stack (accessed on 23/10/2023)
78. Apache, "Kafka 3.0 Documentation," Available on-line at: https://kafka.apache.org/documentation.htm (Access on 23/10/2023)
79. PDMFC, "CHIMERA—Anonymization Framework," Available on-line at: https://pdmfc.com/bias.html?key=chimera (Access on 23/10/2023)
80. MISP, Available on-line at: https://www.misp-project.org/ (Access on 23/10/2023)
81. AEGIS IT Research, "AEGIS Forensics Visualization Toolkit (FVT)". Available on-line at: https://aegisresearch.eu/solutions/forensics-visualization-toolkit-fvt/ (Access on 23/10/2023)
82. Islam, S., Grigoriadis, C., Papastergiou, S. Information sharing for creating awareness for securing healthcare ecosystem. 19th International Conference on the Design of Reliable Communication Networks (DRCN), IEEE, Vilanova i la Geltru, Spain, pp. 1–5 (2023)
83. Cho, S., et al., Cyber kill chain based threat taxonomy and its application on cyber common operational picture. International Conference on Cyber Situational Awareness, Data Analytics, and Assessment (Cyber SA 2018), June 2018, Glasgow, UK.
84. Montesino, R., et al.: SIEM-based framework for security controls automation. Inform Manage Comput Secur Emerald **20**(4), 248–263 (2012)
85. Zamfir, V.A., Carabas, M., Carabas, C., Tapus, N.: Systems monitoring and big data analysis using the Elasticsearch system. Int Conf Control Syst Comput Sci (CSCS), IEEE (2019). https://doi.org/10.1109/CSCS.2019.00039
86. Cisco and Sourcefire, "Snort IPS tool". Available on-line at: https://www.snort.org/ (Access on 23/10/2023).
87. Kioskli, K., et al.: The importance of conceptualising the human-centric approach in maintaining and promoting cybersecurity-hygiene in healthcare. Applied Sciences MDPI **13**(6), 1–16 (2023)
88. Cichonski, K.S.P., Millar, T., Grance, T.: Computer security incident handling guide: recommendations of the national institute of standards and technology," NIST Spec. Publ., vol. 800–61, p. 79, 2012, [Online]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-61r2.pdf (Access on 23/10/2023)
89. Garzón, F.: Cybersecurity incident response.4 (2020)
90. Ahmad, A., Desouza, K.C., Maynard, S.B., Naseer, H., Baskerville, R.L.: How integration of cyber security management and incident response enables organizational learning. J. Assoc. Inf. Sci. Technol. **71**(8), 939–953 (2020). https://doi.org/10.1002/asi.24311
91. A. Castiglione, et al., (2010) An enhanced firewall scheme for dynamic and adaptive containment of emerging security threats. Proc2010 Int Conf Broadband Wirel Comput Commun Appl Bwcca 475–481 https://doi.org/10.1109/BWCCA.2010.117
92. Adamov, A., Carlsson, A.: Cloud incident response model. Proc. 2016 IEEE East-West Des. Test Symp. EWDTS 2016 1–3 (2016) https://doi.org/10.1109/EWDTS.2016.7807665.
93. Baliga, A. Chen, X. Iftode, L.: Paladin: automated detection and containment of rootkit attacks. 20, (2014) [Online]. Available: https://pdfs.semanticscholar.org/f51f/9be6b02d2c2ec2a414a14dde4979765f6670.pdf (Access on 23/10/2023).
94. Ceron, J.M., Margi, C.B., Granville, L.Z.: MARS: from traffic containment to network reconfiguration in malware-analysis systems. Comput. Networks **129**, 261–272 (2017). https://doi.org/10.1016/j.comnet.2017.10.003
95. Lamis, T.: A forensic approach to incident response. Proc. 2010 Inf Secur Curric Dev Annu Conf InfoSecCD **10**, 177–185 (2010). https://doi.org/10.1145/1940941.1940975
96. CONCORDIA EU project, 2019–2022. Available on-line at: https://www.concordia-h2020.eu/ (accessed on 23/10/2023).
97. Chou, D., Jiang, M.: A survey of data-driven network intrusion detection. ACM Comput Surv ACM **54**(9), 1–36 (2021)
98. Jain, R.:WUSTL EHMS 2020 dataset for internet of medical things (IoMT) cybersecurity research. Washington University in St. Louis, 2020. Available on-line at: https://www.cse.wustl.edu/~jain/ehms/index.html (accessed on 23/10/2023)

99. Lippmann, R., Haines, J., Fried, D., Korba, J., Das, K.: The 1999 DARPA offline intrusion detection evaluation. Comput Networks Elsevier **34**(2000), 579–595 (2000)
100. Sigholm, J., Bang, M.: Towards Offensive Cyber Counterintelligence: Adopting a Target-Centric View on Advanced Persistent Threats. Intell Secur Inform Conf (EISIC), 2013 European IEEE (2013). https://doi.org/10.1109/EISIC.2013.37
101. Vazquez, D. F., Acosta, O. P., Spirito, C., Brown, S., Reid, E., Conceptual framework for cyber defense information sharing within trust relationships. 4th International Conference on Cyber Conflict, CyCon 2012, Tallinn, Estonia, June 5–8, (2012) 2012, 1–17
102. Taddeo, M., McCutcheon, T., Floridi, L.: Trusting artificial intelligence in cybersecurity is a double-edged sword. Nat Mach Intell Springer Nat **1**(12), 557–560 (2019)
103. Hatzivasilis, G. et al., Secure semantic interoperability for IoT applications with linked data. IEEE Global Communications Conference (GLOBECOM 2019), IEEE, Waikoloa, HI, USA, 9–13 December, (2019) pp. 1–7.
104. Krasznay, C. Gyebnar, G. Possibilities and limitations of cyber threat intelligence in energy systems. 13th International Conference on Cyber Conflict, NATO CCDCOE Publications, Talin, Estonia, (2021) pp. 171–188
105. Ring, T.: Threat intelligence: why people don't share. Comput Fraud Secur Elsevier **2014**(3), 5–9 (2014)
106. Guo, L. et al., (2021) Overview of cyber threat intelligence description. International Conference on Applications and Techniques in Cyber Intelligence (ATCI), Fuyang, China, Springer AISC 1398: 343–350
107. Macnish, K., FernandezInguanzo, A., Kirichenko, A.: Smart information systems in cybersecurity. ORBIT J **2**(2), 1–26 (2019)
108. Kioskli, K., Mouratidis, H., Polemi, N.: Bringing humans at the core of cybersecurity: Challenges and future research directions. Human Factors Cybersecurity AHFE Open Access **91**, 82–92 (2023)
109. Kioskli, K., Dellagiacoma, D., Fotis, T., Mouratidis, H.: The supply chain of a Living Lab: Modelling security, privacy, and vulnerability issues alongside with their impact and potential mitigation strategies. J Wirel Mob Networks Ubiquitous Comput Depend Appl **13**(2), 147–182 (2022)
110. Frank, L., et al.: Policy-based identification of IoT devices' vendor and type by DNS traffic analysis. Policy-Based Auton Data Govern Springer LNISA **11550**, 180–201 (2019)
111. Jiang, H., Bouabdallah, A.: JACPoL: a simple but expressive JSON-based access control policy language. 11th IFIP International Conference on Information Security Theory and Practice, IFIP, (2017) Heraklion, Crete, Greece 56–72
112. Ahmed, A.J. et al., Policy-based QoS management framework for software-defined networks. International Symposium on Networks, Computers and Communications (ISNCC), 1–7 (2018)

Authors and Affiliations

George Hatzivasilis[1] · Eftychia Lakka[1] · Manos Athanatos[1,2] · Sotiris Ioannidis[2] · Grigoris Kalogiannis[2] · Manolis Chatzimpyrros[2] · George Spanoudakis[2] · Spyros Papastergiou[3] · Stylianos Karagiannis[4] · Andreas Alexopoulos[5] · Dimitry Amelin[6] · Stephan Kiefer[6]

✉ George Hatzivasilis
hatzivas@ics.forth.gr

Eftychia Lakka
elakka@ics.forth.gr

Manos Athanatos
athanat@ics.forth.gr

Sotiris Ioannidis
sotiris@ics.forth.gr

Grigoris Kalogiannis
g.kalogiannis@sphynx.ch

Manolis Chatzimpyrros
m.chatzimpyrros@sphynx.ch

George Spanoudakis
spanoudakis@sphynx.ch

Spyros Papastergiou
spyros.papastergiou@focalpoint-sprl.be

Stylianos Karagiannis
Stylianos.karagiannis@pdmfc.com

Andreas Alexopoulos
andreas.alexopoulos@aegisresearch.eu

Dimitry Amelin
dmitry.amelin@ibmt.frauhofer.de

Stephan Kiefer
stephan.Kiefer@ibmt.frauhofer.de

[1] Institute of Computer Science, Foundation for Research and Technology—Hellas (FORTH), Heraklion, Greece

[2] Innovation Department, Sphynx Technology Solutions AG, Zug, Switzerland

[3] Innovation Department, Focal Point, Brugge, Belgium

[4] Innovation Department, PDMFC, Lisbon, Portugal

[5] Innovation Department, Aegis It Research GmbH, Brunswick, Germany

[6] Fraunhofer Gesellschaft Zur Foerderung Der Angewandten Forschung E.V., Munich, Germany