Cyber LOPA: An Integrated Approach for the Design of Dependable and Secure Cyber-Physical Systems

Ashraf Tantawy, *Member, IEEE*, Sherif Abdelwahed, *Senior Member, IEEE*, and Abdelkarim Erradi, *Member, IEEE*

***Abstract*—Safety risk assessment is an essential process to ensure a dependable cyber-physical system (CPS) design. Traditional risk assessment considers only physical failures. For modern CPSs, failures caused by cyber attacks are on the rise. The focus of latest research effort is on safety–security lifecycle integration and the expansion of modeling formalisms for risk assessment to incorporate security failures. The interaction between safety and security lifecycles and its impact on the overall system design, as well as the reliability loss resulting from ignoring security failures, are some of the overlooked research questions. This article addresses these research questions by presenting a new safety design method named cyber layer of protection analysis (CLOPA) that extends the existing layer of protection analysis (LOPA) framework to include failures caused by cyber attacks. The proposed method provides a rigorous mathematical formulation that expresses quantitatively the tradeoff between designing a highly reliable and a highly secure CPS. We further propose a co-design lifecycle process that integrates the safety and security risk assessment processes. We evaluate the proposed CLOPA approach and the integrated lifecycle on a practical case study of a process reactor controlled by an industrial control testbed and provide a comparison between the proposed CLOPA and current LOPA risk assessment practice.**

***Index Terms*—Cyber-physical system (CPS), hazard and operability (HAZOP), IEC 61511, layer of protection analysis (LOPA), NIST SP 800-30, risk assessment, supervisory control and data acquisition, safety instrumented system (SIS), safety integrity level (SIL), security.**

I. Introduction

A CYBER physical system (CPS) is an integration of a physical process with computation and networking required for physical system monitoring and control. The integration of process dynamics with those of computation and networking brings a plethora of engineering challenges. As the majority of CPSs are deployed in mission-critical applications, the dependability and resilience to failures is a key design property for modern CPS.

To ensure that a given CPS is dependable, a risk assessment is carried out at both design time and operation time. The risk assessment process highlights the system weaknesses and helps define the safety requirements that need to be met to achieve the target reliability measures. The classical approach to perform the risk assessment is to consider physical failures only. As state-of-the-art CPS designs move to open-source hardware and software, cyber attacks have become a source of failure that cannot be ignored.

Realizing the critical nature of CPS cyber attacks and their impact on the safety of people and environment, as well as the potential catastrophic financial losses, the research community developed several approaches to integrate security aspects into the safety risk assessment process. This integration has been done mainly by extending the reliability modeling formalism to incorporate security-related risks. One of the overlooked research questions is how safety and security interact with each other and how this interaction would impact the overall system design. Putting this research question in a different format: *Is there a tradeoff between designing a highly reliable and a highly secure system?* a related research question is: *If we ignore the cyber security attacks in the design process, what is the impact on the overall system reliability? Is the reliability gain worth the complexity introduced by integrating security both at design and runtime?* A follow-up research question is: *Under what conditions can we ignore security failures?*

In order to better understand the interaction between safety and security lifecycles in the system design process, we consider in this article the safety risk assessment process and study the impact of overlooking failures caused by cyber attacks. We refer to such failures as security failures in the rest of this article. By formally introducing the failures caused by attacks into the risk assessment process, we can define the reliability requirements for the cyber components of the system as a function of both the failure rate of physical components and the resilience to cyber attacks. This formal requirement specification enables us to understand the design tradeoff between higher reliability of physical components versus higher resilience of cyber components, and the sensitivity of the overall system performance to both types of failures. In addition, we can gain insight into the interplay between safety and security and how to integrate both lifecycles

Manuscript received February 24, 2021; revised September 22, 2021 and January 16, 2022; accepted March 11, 2022. Date of publication April 22, 2022; date of current version June 2, 2022. This work was supported by the Qatar National Research Fund (a member of the Qatar Foundation) under Grant NPRP 9-005-1-002. The statements made herein are solely the responsibility of the authors. Associate Editor: W. Dong. *(Corresponding author: Ashraf Tantawy.)*

Ashraf Tantawy is with the School of Computer Science and Informatics, De Montfort University, LE1 9BH Leicester, U.K. (e-mail: ashraf.tantavy@dmu.ac.uk).

Sherif Abdelwahed is with the Department of Electrical and Computer Engineering, Virginia Commonwealth University, Richmond, VA 23284 USA (e-mail: sabdelwahed@vcu.edu).

Abdelkarim Erradi is with the Department of Computer Science and Engineering, Qatar University, Doha 2713, Qatar (e-mail: erradi@qu.edu.qa).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TR.2022.3163652.

Digital Object Identifier 10.1109/TR.2022.3163652

during the design process. More specifically, we consider the layer of protection analysis (LOPA), a widely adopted risk assessment method that follows a hazard identification study, such as hazard and operability (HAZOP). LOPA is carried out to identify whether an additional safety instrumented system (SIS) is needed for specific hazardous scenarios to achieve the target risk level. As a modern SIS is typically an embedded device, it has both physical and security failure modes. We mathematically derive the SIS design constraints in terms of both physical and security failure probabilities. Additionally, we propose an integrated safety–security design process that shows the flow of information between both lifecycles.

We can classify the research work on combining safety and security for CPSs into two broad categories that try to answer the following research questions: 1) given the independent safety and security lifecycles, what are the similarities/differences and how could the two lifecycles be aligned or unified? This research direction usually focuses on answering the question "what to do," rather than "how to do it"; and 2) for a given CPS, how can we carry out risk assessment (qualitative/quantitative) that considers both physical failures and cyber attacks? Consequently, how can we unify the process of safety and security requirements definition and verification? This research direction focuses on common modeling techniques that can incorporate both safety and security failures and often extends model-based engineering body of knowledge and tools to incorporate security requirements in the design process. In Section VI, we survey the main results for each research direction. A more thorough survey is presented in [1] and [2].

Our contribution: The work presented in this article addresses both research directions with a new approach. First, we integrate both safety and security lifecycles based on a rigorous mathematical formulation that captures their interaction. The formulation enables the designer to assess how a security design decision would impact system safety. This is in contrast to the existing research work that does not explicitly model the dynamic safety–security lifecycle interaction. Second, we develop an integrated safety–security design lifecycle and show in detail how to apply it to a real-world design, distinguishing the work from abstract research on risk assessment that does not carry over to the design stage. Finally, our approach is founded on LOPA, a practical approach that is extensively used in industry, giving the approach the merit for industrial implementation.

The rest of this article is organized as follows. Section II introduces the background information required for problem setup, including IEC 61511 safety lifecycle and the LOPA method, cyber dependence between control and safety systems, and cyber security risk assessment. Section III proposes a new LOPA mathematical formulation called cyber layer of protection analysis (CLOPA) that incorporates failures due to cyber attacks. Section IV proposes an integrated safety–security lifecycle process. Section V presents a case study for the design of a safety system for a chemical reactor, comparing classical LOPA approach to the proposed CLOPA formulation. Section VI summarizes the related work on safety–security co-design. Finally, Section VII concludes this article.

II. Safety and Security Risk Assessment

There are two main embedded systems that control and safeguard a given physical system: the control system and the safety system. In the process industry, the control system is referred to as the basic process control system (BPCS), and the safety system is referred to as the SIS. In practice, both the systems typically have a programmable controller architecture with one or more back planes, processor cards, and a variety of input–output interface cards [3]. For larger systems, BPCS and SIS architectures comprise multiple distributed nodes connected via a communication backbone. Fig. 2 depicts the two systems and their connectivity over a control network. In the following, we briefly discuss the SIS design lifecycle, BPCS and SIS security lifecycles, and their interaction.

A. IEC 61511 Safety Lifecycle Process

Fig. 1 shows the SIS design lifecycle according to IEC 61511 standard [4]. The design starts with hazard and risk assessment, where systems hazards are identified. HAZOP study, what if analysis, and fault tree analysis are the most common methods at this stage [5]. The risk assessment phase ranks each identified risk according to its likelihood and consequence, either quantitatively or qualitatively, and associates a risk ranking for each hazard. The resulting list of hazards and associated risk ranking is used as an input to the second phase focused on the allocation of safety functions to protection layers. This phase deals only with hazards that exceed a threshold risk rank that an organization is willing to accept. For each hazardous scenario, there is a target mitigated event likelihood (TMEL) measure that is defined based on the risk rank. The purpose of this phase is to check if the TMEL is met with existing protection layers. If not, an additional protection layer is recommended, often in the form of a new safety instrumented function (SIF) with a predefined safety integrity level (SIL) to cover the gap to the TMEL. The SIF comprises one or more sensors, a logic solver, and one or more actuators. The logic solver is commonly referred to as the SIS. An example SIF is illustrated in Fig. 6 for an overflow hazardous scenario of a reactor system, which will be discussed in detail in Section V. Risk matrix, risk graph, and LOPA are the most commonly used methods for the allocation of safety functions to protection layers [3].

The third phase is the development of the safety requirement specification (SRS), which documents all the functional and timing requirements for each SIF. The fourth phase is the detailed design and engineering. Phases 5–8 are concerned with system installation and commissioning, operation, modification, and decommissioning. Phase 2 is where the CPS control and safety systems are considered in the risk assessment process. Therefore, we study this phase in depth in this article. Since LOPA is the predominant approach for this phase, we limit our discussion to LOPA methodology. Other approaches could be adopted in a similar way.

The underlying assumption in LOPA analysis is that all protection layers, including the new SIF, are independent. In other words, if one layer failed, this does not increase or decrease

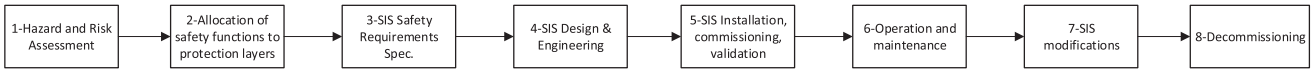

Fig. 1. IEC 61511 SIS design lifecycle (adopted from [4]).

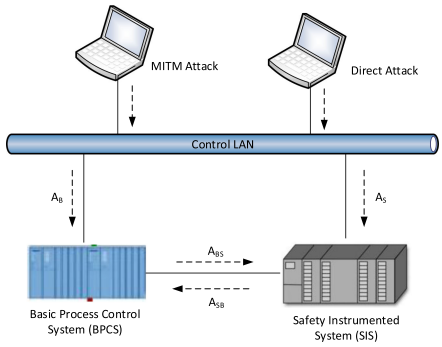

Fig. 2. Snapshot of an industrial control system architecture showing BPCS–SIS connectivity and potential attack vectors.

the likelihood of failure of the other layers. This assumption simplifies the mathematical analysis significantly, as it allows the multiplication of individual probabilities to obtain the required joint probability. The simplicity of LOPA calculations is probably one of the key reasons behind its widespread adoption by industry. Unfortunately, when cyber security is considered as a potential failure in LOPA analysis, the independence assumption between the control and safety systems no longer holds, as explained in the next section.

B. Control and Safety System Cyber Dependence

Each of the control and safety systems has two modes of failure: BPCS physical failures, B_p, BPCS security failure, B_c, SIS physical failure, S_p, and SIS security failure S_c. For physical failures, IEC 61511 standard strongly recommends complete separation between the control and safety systems of any plant. This separation includes sensors, computing devices, and final elements such as valves and motors. Separation also includes any common utility such as power supplies. The industry adopted this separation principle; hence, BPCS and SIS physical failures could be accurately assumed to be independent, i.e., $P[B_p, S_p] = P[B_p]P[S_p]$.

One exception to the separation between BPCS and SIS is the cyber communication link between the control and safety systems. Fig. 2 shows a snapshot of a typical industrial control system architecture showing the communication link between the BPCS and the SIS. BPCS–SIS communication could be over the control LAN or via a dedicated point to point serial link. The communication protocol is typically an open standard such as Modbus or DNP3 [6], [7]. This type of communication exists in many industrial installations to exchange plant data, as the data from field devices connected to the safety system are not accessible from the BPCS and *vice versa*. Given this architecture, we can define two attack vectors for SIS compromise: 1) a direct attack that exploits an existing controller vulnerability could be launched against the SIS node. This could be via any node on the control LAN or using man-in-the-middle (MITM) attack that exploits the BPCS–SIS communication. We designate this attack event by A_S in Fig. 2; and 2) by compromising the BPCS first and then exploiting the BPCS–SIS link to compromise the SIS. We designate this pivot attack by the sequence of events A_B and A_{BS} in Fig. 2. Furthermore, we designate the attack event from the SIS to the BPCS by A_{SB}. The attack sequence $A_B \rightarrow A_{BS}$ may be easier if the SIS is highly secured such that a direct attack may be infeasible. This is particularly true if we consider the fact that the BPCS is a trusted node to the SIS.

The above analysis shows a clear dependence between the control and safety systems that violates the original LOPA independence assumption. The security failures for the BPCS and the SIS are no longer independent because of the data communication coupling. We can formulate the different security failure probabilities as in (1)–(3) using basic probability laws with the aid of Fig. 2, where $P[A_i]$ is interpreted as the probability of success of attack A_i. Furthermore, the considered attack A_i should have the impact of stopping the BPCS or SIS from performing its intended control or safeguard function as related to the hazard under study. This is important because not all attacks that exploit controller vulnerabilities result in a process hazard. Therefore, the attacks considered represent a subset of the complete set of attacks that could exploit the BPCS or SIS vulnerabilities. Accordingly, from hereafter, $P[A_B]$, $P[A_S]$, $P[A_{BS}]$, and $P[A_{SB}]$ refer to the relevant attacks that cause a process hazard. This concept is revisited throughout this article and is made more clear in the case study when sample attacks are presented. For a case study example on the fact that not all cyber failures have a system reliability consequence, we refer the interested reader to [8] for a study on the impact of different software failure modes on system reliability for the electric power grid domain. Finally, it can be easily shown that if the BPCS–SIS communication link does not exist, or fully secured, then $P[A_{SB}] = P[A_{BS}] = 0$, and (3) reduces to the independent case $P[S_c, B_c] = P[S_c]P[B_c]$

$$P[B_c] = P[A_B] + P[A_S]P[A_{SB}] - P[A_B]P[A_S]P[A_{SB}] \tag{1}$$

$$P[S_c] = P[A_S] + P[A_B]P[A_{BS}] - P[A_B]P[A_S]P[A_{BS}] \tag{2}$$

$$P[S_c, B_c] = P[A_B](P[A_S] + P[A_{BS}]) + P[A_S]P[A_{SB}] - P[A_B]P[A_S](P[A_{BS}] + P[A_{SB}]). \tag{3}$$

C. Cyber Security Risk Assessment

The calculation of the probability of cyber attacks A_S, A_B, and A_{BS} could be performed during the cyber security risk assessment process. This requires a detailed specification of

the BPCS and the SIS and their connectivity, including the embedded system hardware, operating system, running software services, and the network connectivity. According to the National Institute of Standards and Technology (NIST) SP 800-30 standard, "Guide for Conducting Risk Assessments," the cyber security lifecycle process stages are: 1) asset identification, where the particular cyber components and their criticality levels are identified; 2) vulnerability identification, along with the associated threats and attack vectors; 3) the development of relevant attack trees for each attack scenario identified; 4) penetration testing to validate the vulnerability findings and attack scenarios and to help estimating the effort and probability for individual attack steps for each scenario; and 5) risk assessment to identify the scenarios with unacceptable risk [9]. Fig. 5 shows the BPCS and SIS cyber security lifecycles. In this article, we follow the same cyber security lifecycle, but with the physical process as the main focus. Therefore, for asset identification, the cyber component criticality is primarily identified by its failure impact on the operation of the connected physical component. Similarly, for vulnerability identification, threats and attack vectors are filtered by their impact on the physical process. Attacks that do not disturb the controlled process are ignored as they have no direct impact on the process safety. In addition, such impacts take place with much higher probability at the corporate network level, so they can be ignored with minimum impact on the risk assessment at the control network level. For more detailed discussion on process-driven attack identification, we refer the reader to [10].

For the presented architecture, the BPCS and the SIS are the critical components in direct contact with the process. The calculation of the required BPCS and SIS security failure probabilities could be typically carried out with the aid of attack trees [11]. The attack tree enumerates all the possible routes to compromise the system, and each edge is assigned a probability representing the likelihood of the associated event. Using basic probability laws, the overall probability of a system compromise could be calculated. Section V presents an example of such calculation. We re-emphasize the fact that the scope of cyber security risk assessment and attack trees in this case will be limited to attacks targeting the physical process to cause a process hazard. Although information security attacks with objectives such as stealing information are possible, most of this information is already available at the corporate network level, and an attacker who penetrates down to the control network level to compromise an BPCS or SIS will conceivably have the goal of physical process attack.

III. CLOPA: LOPA With Security Failures

A. Mathematical Formulation

In risk assessment, an initiating event is an unplanned event that when occurring may lead to a hazard. Examples of initiating events include equipment failure, human error, and cyber attacks. A system hazard will take place if one or more of the initiating events occur, and all the associated protection layers against that hazard fail simultaneously. The main objective of LOPA is to calculate the expected number of hazardous events per time interval and compare it to the TMEL. We designate the random variable representing the number of events per unit time for a specific initiating event by N, the random variable representing the simultaneous failure of protection layers when the initiating event occurs by L, where L is Bernoulli distributed with success probability p, and the random variable representing the number of hazards per time interval by H. We then have

$$H = \sum_{i=1}^{N} \mathbb{I}_E(l_i) \tag{4}$$

where $\mathbb{I}$ is the indicator function, and the set $E = \{l : l_i = 1, i = 1:N\}$. We note that for a given $N = k$, H is a binomially distributed random variable with expected value $E[H|N = k] = kp$. Therefore

$$\begin{aligned} E[H] &= \sum_{k=0}^{\infty} E[H|N = k]P[N = k] \\ &= p\sum_{k=0}^{\infty} kP[N = k] = pE[N] = p\lambda \end{aligned} \tag{5}$$

where λ represents the expected value of the number of initiating events per unit time, N. Although N is typically modeled by a Poisson random variable in reliability engineering, we do not assume any specific distribution in the analysis. This is particularly important because some initiating events considered in the article, such as security failures, are not accurately modeled by a Poisson distribution.

Equation (5) is the underlying mathematical concept behind LOPA analysis. Essentially, for each initiating event, the likelihood λ is estimated from field data, and the probability of simultaneous failure of all protection layers is specified. Finally, the expected number of hazards per unit time, $E[H]$, considering all initiating events, is estimated and compared to the prespecified TMEL. If $E[H] >$ TMEL, then a SIS is required with a probability of failure on demand $P[S_p]$ (or equivalently a risk reduction factor (RRF) $= 1/P[S_p]$) that achieves $E[H] \leq$ TMEL.

In order to express the LOPA formula in (5) in terms of all protection layers, including the BPCS and the SIS, we introduce some mathematical notation. We designate the set of initiating events for a given hazardous scenario by $\mathcal{I} = \{I_1, I_2, \ldots, I_n, B_p, \mathcal{A}_r\}$, where n is the number of possible initiating events excluding BPCS failures, B_p denotes the BPCS physical failure event, and $\mathcal{A}_r$ denotes the set of attacks relevant to the hazard under study. We express the associated set of event likelihoods by $\Lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_n, \lambda_p, \lambda_c\}$. Furthermore, we denote the set of all possible protection layers by $\mathcal{L} = \{L_1, L_2, \ldots, L_m\}$, where m is the number of existing protection layers, excluding the BPCS and the SIS. BPCS protection is denoted by B, and SIS protection is denoted by S. For each initiating event i, there is a subset of protection layers $\mathcal{L}_i \subseteq \mathcal{L} \cup \{B, S\}$ that could stop the propagation of a hazard from causing its consequences. Table I shows a sample LOPA table using the introduced terminology.

TABLE I
SAMPLE LOPA TABLE

Initiating Event	Likelihood λ_i (/yr)	L_1 ... L_m	BPCS (B)	TMEL
I_1	λ_1	$\leftarrow P[\mathcal{L}_1] \rightarrow$	$P[B]$	10^{-x}
...	...		...	
I_n	λ_n	$\leftarrow P[\mathcal{L}_n] \rightarrow$	$P[B]$	
B_p	λ_p	$\leftarrow P[\mathcal{L}_B] \rightarrow$	1	
$\mathcal{A}_r$	λ_c	$\leftarrow P[\mathcal{L}_B] \rightarrow$	$P[B]$	

$P[\mathcal{L}]$ refers to the combined probability of failure of protection layers applicable to the initiating event from L_1 to L_m.

B. Semantically Relevant Attack Event Formulation

We designate the set of all possible attacks against the BPCS by $\mathcal{A}$. For a given hazard under consideration, the subset of attacks that would lead to this hazard, i.e., the contextually or semantically relevant attacks, is designated by $\mathcal{A}_r$. To estimate the likelihood of relevant cyber attacks (expected number per unit time), λ_c, we assume an attacker profile with an average rate of launching attacks per unit time λ. For every launched attack, the attacker is presented with the complete set of attacks $\mathcal{A}$ and selects only one attack a with probability $P[A = a] = \alpha_a$ that is dependent on the attacker profile, such that

$$\sum_{a \in \mathcal{A}} P[A = a] = \sum_{a \in \mathcal{A}} \alpha_a = 1. \tag{6}$$

The likelihood of cyber attack $a \in \mathcal{A}_r$ is then $\lambda_a = \lambda \alpha_a$. This cyber attack has the potential to cause a system hazard if both the BPCS and the SIS fail jointly to stop the attack (either physical or cyber failure). We designate this probability by $P_a[S, B]$. The exact approach to include every attack $a \in \mathcal{A}_r$ in the LOPA table is to treat each attack as an individual entry, akin to the last row in Table I, with initiating event likelihood λ_a. However, this approach has two main drawbacks: First, the number of attacks could be large, and this would grow the LOPA sheet significantly. Second, the treatment of each attack individually would not allow us to utilize attack modeling techniques such as attack trees that model collectively all the possible attack paths for one attack objective. Therefore, we adopt an alternative approach, where all relevant attacks $a \in \mathcal{A}_r$ could be represented by one entry in the LOPA table. The following lemma summarizes the approximate solution. The proof is included in the Appendix.

Lemma 3.1: Assume a given hazard scenario H, a control system BPCS, an average rate of launching attacks against BPCS λ, Hazard H semantically-relevant attack set $\mathcal{A}_r$, and probability α_a of selecting attack $a \in \mathcal{A}_r$. Then, the impact of all initiating events $a \in \mathcal{A}_r$ on the LOPA calculation could be approximated by a single initiating event with likelihood $\lambda_c = \lambda \sum_{a \in \mathcal{A}_r} \alpha_a$ and a BPCS failure probability with respect to the combined set of attacks $a \in \mathcal{A}_r$, where each attack probability is weighted by the factor $\gamma_a = \alpha_a / \sum_{a \in \mathcal{A}_r} \alpha_a$.

The lemma enables us to use attack trees with leaf nodes weighted by γ_a to calculate the BPCS security failure probability in response to the combined set of attacks $\mathcal{A}_r$ with likelihood $\lambda \sum_{a \in \mathcal{A}_r} \alpha_a$. For the special case where the cyber attacker profile results in random selection of the attack $a \in \mathcal{A}$, e.g., an attacker with no knowledge about the system, the likelihood reduces to $\lambda |\mathcal{A}_r| / |\mathcal{A}|$ and the leaf node weights reduce to $\gamma_a = 1/|\mathcal{A}_r| \, \forall a$. We use this special case in the case study in Section V.

C. Cyber LOPA Formulation

With the introduced notation, the expected number of hazards in (5), which should be less than the TMEL, could be expanded as

$$E[H] = P[S, B] \left(\sum_{i=1}^{n} (\lambda_i P[\mathcal{L}_i]) + \lambda_c P[\mathcal{L}_B] \right) + \lambda_p P[S] P[\mathcal{L}_B] \leq \text{TMEL} \tag{7}$$

where $\mathcal{L}_B$ is the set of protection layers for BPCS physical or security failure event, and we assume that all protection layers are independent of the BPCS and the SIS, while keeping the dependence between the BPCS and the SIS. In addition, higher order probability terms resulting from multiple initiating events are ignored due to their insignificance.

To calculate the joint failure probability $P[S, B]$, we use basic probability laws and the fact that the BPCS and the SIS have both the physical and cyber modes of failure, as explained in Section II-B, to obtain

$$\begin{aligned} P[S, B] = {} & P[S_p] \left(P[B_p](1 - P[B_c] - P[S_c]) + P[B_c] \right) \\ & + P[S_c, B_c] \left(1 - P[S_p] - P[B_p] + P[S_p] P[B_p] \right) \\ & + P[S_c] P[B_p]. \end{aligned} \tag{8}$$

Substituting (8) into (7), we obtain the general LOPA equation in (9). We call this expanded version of LOPA hereafter CLOPA

$$P[S_p] \leq \frac{\beta - (\alpha_1 P[S_c] + \alpha_2 P[S_c, B_c])}{\alpha_1 - \alpha_1 P[S_c] + \alpha_2 P[B_c] - \alpha_2 P[S_c, B_c]} \tag{9}$$

where

$$\alpha_1 = P[B_p] \left(\sum_{i=1}^{n} (\lambda_i P[\mathcal{L}_i]) + \lambda_c P[\mathcal{L}_B] \right) + \lambda_p P[\mathcal{L}_B] \tag{10}$$

$$\alpha_2 = (1 - P[B_p]) \left(\sum_{i=1}^{n} (\lambda_i P[\mathcal{L}_i]) + \lambda_c P[\mathcal{L}_B] \right) \tag{11}$$

$$\beta = \text{TMEL}. \tag{12}$$

In order to define the CLOPA formula in terms of the actual design variables $P[A_S]$ and $P[A_{BS}]$ that represent the probability of security failures of actual CPS components, we substitute (1)–(3) into (9) to obtain

$$P[S_p] \leq \frac{\beta - \gamma_1 P[A_S] - \gamma_2 P[A_{BS}](1 - P[A_S])}{\gamma_3 - \gamma_3 P[A_S] - \gamma_2 P[A_{BS}](1 - P[A_S])} \tag{13}$$

where:

$$\gamma_1 = \alpha_1 + \alpha_2 [P[A_B] + P[A_{SB}](1 - P[A_B])] \tag{14}$$

$$\gamma_2 = (\alpha_1 + \alpha_2) P[A_B] \tag{15}$$

$$\gamma_3 = \alpha_1 + \alpha_2 P[A_B]. \tag{16}$$

Equation (13), along with (10)–(12) and (14)–(16), represents the general CLOPA formulation to design the SIS. It represents

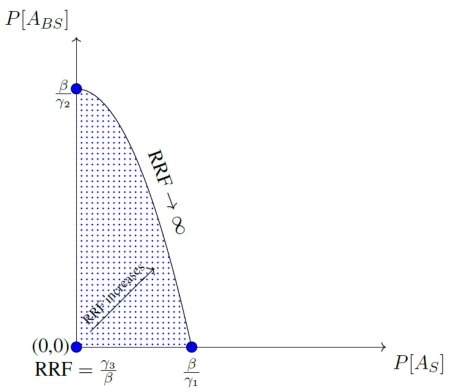

Fig. 3. CLOPA design region (shaded). Any point in the shaded region results in a feasible SIS. Points near the boundary require an SIS with a very high RRF value and, hence, difficult to obtain in practice.

an upper bound on the probability of physical failure for the safety system in terms of the security failure probabilities, showing clearly the coupling between the safety system and security system design. The design of the SIS should satisfy (13), where the design variables are $P[S_p]$, $P[A_S]$, and $P[A_{BS}]$. The rest are model parameters that are predetermined, including the BPCS failure marginal probabilities. This is because the BPCS design is independent of the SIS design and usually takes place earlier in the engineering design cycle. Note that we assume here that $P[A_{SB}]$ is a known parameter. This is because by completely defining the BPCS and its hardware and software specifications, the probability of a cyber attack compromising process safety could be estimated, even though the SIS is not yet completely defined. Table 1 in the Appendix summarizes the model variables, parameters, and how the model parameters are calculated.

It should be noted that with existing LOPA methodology, security failures are ignored, i.e., $P[A_B] = P[A_S] = P[A_{BS}] = P[A_{SB}] = 0$. Substituting these zero values into (13), we obtain the classical LOPA formulation

$$P[S_p] \leq \frac{\beta}{\alpha_1} = \frac{\text{TMEL}}{P[B_p]\sum_{i=1}^{n}(\lambda_i P[\mathcal{L}_i]) + P[\mathcal{L}_B](\lambda_p + \lambda_c)}. \tag{17}$$

D. Design Space

Using the fact that $P[S_p] \geq 0$ for a realizable safety system in (13), we obtain

$$\gamma_1 P[A_S] + \gamma_2 P[A_{BS}] - \gamma_2 P[A_S]P[A_{BS}] \leq \beta. \tag{18}$$

Fig. 3 shows the shaded region defined by the inequality in (18). The boundary curve is defined by (18) when the following equality holds:

$$P[A_{BS}] = \frac{\beta}{\gamma_2}\left(\frac{1 - \left(\frac{\gamma_1}{\beta}\right)P[A_S]}{1 - P[A_S]}\right). \tag{19}$$

The first-order derivative of the boundary curve is negative for $\gamma_1/\beta > 1$ and positive otherwise. Since $\gamma_1 > \beta$ to require a safety system [proof is straightforward by inspecting (10), (11), (14), and (17)], the boundary curve is concave as in Fig. 3. We note that any point in the shaded region results in a feasible SIS. Points on the boundary curve result in $P[S_p] = 0$ or equivalently RRF $\to \infty$. Points closer to the boundary would have high values for the RRF, requiring a very highly reliable SIS that may not be achievable in practice. Points closer to the origin result in lower RRF. It can be easily shown that the contour lines for (13), where $P[S_p] = C$, could be expressed as

$$P[A_{BS}] = \frac{C\gamma_3 - \beta}{\gamma_2(C-1)}\left(\frac{1 - \left(\frac{C\gamma_3 - \gamma_1}{C\gamma_3 - \beta}\right)P[A_S]}{1 - P[A_S]}\right). \tag{20}$$

The contour line that represents the design boundary in Fig. 3 can be derived from (20) by setting $C = 0$.

We can extract the following information from this graph.

1) The maximum probability of security failure for the safety system by directed attacks is β/γ_1. This probability results in an unrealizable safety system, as the required RRF $\to \infty$.
2) The maximum probability of security failure for the safety system by pivot attack via the BPCS is β/γ_2. Likewise, this probability does not result in a realizable safety system.
3) Finally, the minimum value of the RRF is achieved at the origin for a perfectly secured safety system, where $P[A_S] = P[A_{BS}] = 0$, with the RRF given by

$$P[S_p]_{\max} = \frac{\beta}{\gamma_3}, \quad \text{RRF}_{\min} = \frac{\gamma_3}{\beta}. \tag{21}$$

Clearly, points outside the shaded region result in nonrealizable SIS. This result re-emphasizes the interplay between the safety and security systems of a CPS.

The design space highlights the major difference between LOPA and CLOPA. In LOPA, the SIS requirement is related to reliability in the form of the required SIL. In CLOPA, an additional requirement for the SIS is its security resilience, in the form of an upper bound on the probability of a security failure (cyber attack success), either directly or indirectly via the BPCS.

E. Classical LOPA Error

To obtain the error resulting from using classical LOPA, we subtract (17) from (13) to obtain

$$e_{\text{RRF}} = \frac{\zeta_1 + \zeta_2 P[A_S] + \zeta_3 P[A_{BS}](1 - P[A_S])}{\beta\left[\beta - \gamma_1 P[A_S] - \gamma_2 P[A_{BS}](1 - P[A_S])\right]} \tag{22}$$

where

$$\zeta_1 = \beta(\gamma_3 - \alpha_1) = \beta\alpha_2 P[A_B] \tag{23}$$

$$\zeta_2 = \alpha_1\gamma_1 - \beta\gamma_3 \tag{24}$$

$$\zeta_3 = \gamma_2(\alpha_1 - \beta). \tag{25}$$

The minimum error occurs for a perfectly secured safety system, i.e., $P[A_S] = P[A_{BS}] = 0$:

$$\min e_{\text{RRF}} = P[A_B]\left(\frac{\alpha_2}{\beta}\right). \tag{26}$$

The error will be zero, i.e., classical LOPA result matches CLOPA, if the probability of BPCS security failure via a direct attack is zero.

IV. Safety–Security Co-design

A. Design Process

The current industrial practice is to perform safety and security risk assessments independently, treating the physical and cyber components of a CPS as two separate entities. As illustrated in Section III, accurate safety risk assessment requires knowledge about the cyber components and their security failure probabilities. Formally, the objective is to design an SIS architecture $\mathcal{A}$ that satisfies (13) in terms of both physical and security failure probabilities. Suppose that the architecture $\mathcal{A}$ could be represented by a set of design variables represented by the vector $\mathbf{x}$. If we can relate the physical and security failure probabilities to the vector $\mathbf{x}$ by $P[A_S] = f(x), P[A_{BS}] = g(x), P[S_p] = h(x)$, then we can use these functions to substitute the relevant probabilities into (13), and our design problem will be to find a set of values for the vector $\mathbf{x}$ that satisfies the CLOPA constraint (13). Unfortunately, this design approach is not followed by industry for several reasons. First, abstracting a given architectural design $\mathcal{A}$ into a set of design variables is a very difficult task, not to mention that these design variables have to be linked to both physical and security failures. Second, finding an exact or approximate representation of the functions $f(.), g(.)$, and $h(.)$ that relate the failure probabilities to the design variables may not be possible, as it is not always clear how a design decision would result in a higher or lower probability of failure. Finally, even if we were able to make a perfect modeling, the resulting problem to solve may turn into a discrete optimization problem that is not possible to solve in polynomial time.

Owing to these modeling limitations, the current industrial practice to design SISs (excluding cyber attacks) is to follow an iterative process and rely on engineering judgment during the design process. More precisely, the required risk reduction factor RRF_d is initially calculated; then, the engineering design proceeds to achieve RRF_d using both experience and industrial standard guidelines [4]. After the design is completed, design verification is conducted to calculate the RRF of the proposed design RRF_v. If the resulting $\text{RRF}_v \geq \text{RRF}_d$, then the design stops. Otherwise, the design is refined until the condition $\text{RRF}_v \geq \text{RRF}_d$ is satisfied. In the following, we will adopt the same iterative design approach for CLOPA.

Fig. 4 illustrates the iterative design process. We start with initial values ($P[A_S], P[A_{BS}], \text{RRF}_d$) that satisfy the CLOPA constraint in (13). We then proceed with the SIS design to produce an architecture $\mathcal{A}$. The architecture is then verified to estimate its probability of failure on demand or equivalently its RRF_v. The architecture is also used to carry out a security risk assessment to estimate the probability of security failures $P[A'_S]$ and $P[A'_{BS}]$. If the new set of obtained values ($P[A'_S], P[A'_{BS}], \text{RRF}_v$) still satisfy the CLOPA equation, the design stops. Otherwise, a new iteration will start to adjust the design in order to achieve the CLOPA constraint. This adjustment could be either by adding more security controls or by increasing the reliability of the system using fault-tolerant techniques. Algorithm 1 summarizes the iterative design process.

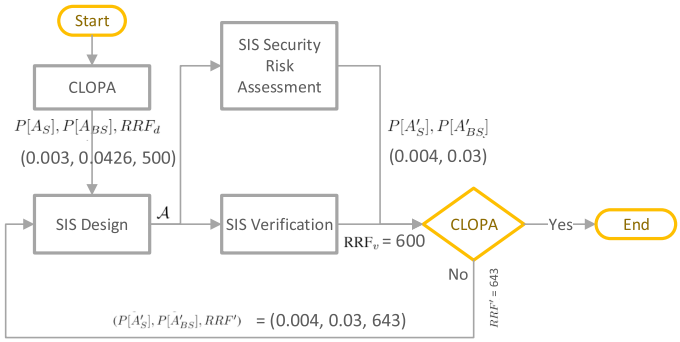

Fig. 4. CLOPA iterative design process. CSTR case study design values are shown.

One question is how can we choose the initial values for RRF , $P[A_S], P[A_{BS}]$? This initial design point could be selected with the aid of the design contour plot as in Fig. 3, where the design point strikes a balance between security and reliability. What is *reasonable* regarding the RRF is well documented in the standards using SILs, and the extra cost to move from one SIL to a higher SIL is well quantified in industry. What is not very clear, though, is what is an achievable value for the probability of security failures. This is still not a well-developed field, and the argument of how to assess such probabilities is still going on in the research community.

Another important question is whether there is any formal guarantees that Algorithm 1 will terminate. To answer this question, we need to know, or at least approximate, how the architectural design $\mathcal{A}$ impacts the RRF, $P[A_S]$, and $P[A_{BS}]$. As pointed out earlier, this is very hard in practice. Without such relationship, the question of convergence to a solution for algorithm termination cannot be precisely answered. However, in practice, modifying the SIS design to increase the RRF is usually done by changing sensor and actuator configuration or reliability figures, as they are often the weakest links in the reliability chain, while the logic solver is minimally changed [4]. Accordingly, for all practical purposes, we can assume that the design process will converge after few runs.

B. Integrated Safety–Security Lifecycle

As the analysis in this article shows a clear coupling between safety and security design requirements, we propose the integrated lifecycle in Fig. 5. In the following, we present a brief description of the lifecycle steps in the order of their execution, according to the numbering labels in Fig. 5.

① *SIS safety lifecycle—HAZOP:* The first step is to carry out the hazard analysis for the physical system, often using HAZOP. This process identifies important assets that may be subject to, or contribute to, risk scenarios. Then, the process identifies all feasible hazards and associated risk ranking, as well as the associated cyber components for each identified hazard. This constitutes an input to the BPCS security lifecycle. If we designate the set of

Algorithm 1: Integrated Safety-Security Lifecycle Design Algorithm.

input : BPCS
output: $\mathcal{A}$, $\boldsymbol{\theta}_S$
$([P[A_B], P[A_{SB}]) \leftarrow$ `BPCS-SecCycle`(BPCS) ;
$\boldsymbol{\theta}_B \leftarrow (P[A_B], P[A_{SB}])$;
$(P[A_S], P[A_{BS}], \text{RRF}_d) \leftarrow$ `DesignContour`$(\boldsymbol{\theta}_B)$;
$\boldsymbol{\theta}_S \leftarrow (P[A_S], P[A_{BS}], \text{RRF}_d)$;
do
  $\mathcal{A} \leftarrow$ `SIS-SafeCycle`$(\boldsymbol{\theta}_S)$;
  $\text{RRF}_v \leftarrow$ `SIS-Verify`$(\mathcal{A})$;
  $(P[A'_S], P[A'_{BS}]) \leftarrow$ `SIS-SecCycle`$(\mathcal{A})$;
  $\text{RRF}' \leftarrow$ `CLOPA`$(P[A'_S], P[A'_{BS}], \boldsymbol{\theta}_B)$;
  $\boldsymbol{\theta}_S \leftarrow \left(P[A'_S], P[A'_{BS}], \text{RRF}'\right)$;
while $RRF_v < RRF'$;
return $\mathcal{A}$, $\boldsymbol{\theta}_S$;

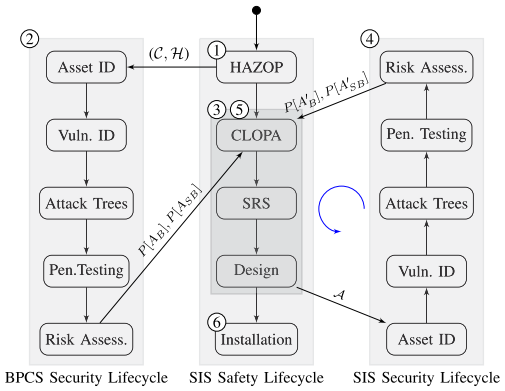

Fig. 5. Integrated safety and security lifecycles. The process starts at ① HAZOP, followed by ② BPCS complete security lifecycle, then ③ SIS safety lifecycle up to the end of the design stage, followed by ④ complete SIS security lifecycle, ⑤, CLOPA check, and possibly several iterations of steps ③, ④, and ⑤ and then terminates at the SIS installation stage.

hazards by $\mathcal{H}$, and the set of cyber components by $\mathcal{C}$, then the output from this process is the function $f : \mathcal{H} \mapsto \mathbb{R}$ representing the risk ranking and the relation $R \subseteq \mathcal{H} \times \mathcal{C}$ representing the cyber components for each hazard.

② *BPCS cyber security lifecycle:* The BPCS cyber security lifecycle, including vulnerability analysis, attack tree generation, penetration testing, and risk assessment, is performed on the BPCS. Ideally, the security risk assessment should be carried out for each process hazard scenario identified during HAZOP to identify the relevant vulnerabilities that may cause a process disruption. However, for the given centralized architecture, the BPCS is typically controlling a large number of control loops; hence, it may not be necessary to repeat the security risk assessment process for each control loop, as vulnerabilities may be applicable to several hazardous scenarios. The output of this process is the BPCS security failure probabilities $P[A_B]$ and $P[A_{SB}]$.

③ *SIS safety lifecycle—CLOPA and SIS design:* The first iteration of CLOPA and SIS design will proceed according to Algorithm 1 and Fig. 4. The CLOPA calculates the design requirement for the SIS in terms of its reliability as defined by the RRF, and its cyber security resilience as defined by $P[A_S]$ and $P[A_{BS}]$. The SIS design then proceeds according to IEC 61511 standard [4] to produce an architecture $\mathcal{A}$. The design includes the hardware architecture, redundancy scheme, and software architecture. The specific design architecture can vary across industries and organizations, but the design has to achieve the required RRF, $P[A_S]$ and $P[A_{BS}]$, as calculated by CLOPA. After the design is completed, SIS verification is carried out to calculate the RRF_v. It should be highlighted that the SIS is one component only of the SIF. The SIF includes the sensor, SIS, and the actuator. Therefore, the verification is carried out on the whole SIF. For a detailed discussion on SIS design and verification, the reader is referred to [3].

④ *SIS cyber security lifecycle:* Using the resulting SIS design hardware and software architecture $\mathcal{A}$, the SIS security lifecycle is carried out. Since the SIS is not yet implemented at this stage, SIS penetration testing is not possible and, hence, omitted from the security lifecycle. The output from this process is the SIS security failure probabilities $P[A'_S], P[A'_{BS}]$, derived from SIS vulnerabilities that may lead to a process hazard. It is noted that the SIS security lifecycle at the right of Fig. 5 proceeds from bottom to top for a better presentation.

⑤ *Safety lifecycle—CLOPA:* The CLOPA calculation is carried out using the values obtained from the safety verification and SIS security lifecycle, $(P[A'_S], P[A'_{BS}], \text{RRF}_v)$, to verify that the architecture $\mathcal{A}$ satisfies the CLOPA constraint. The process SIS safety lifecycle $\rightarrow$ SIS Cyber security lifecycle $\rightarrow$ CLOPA (designated by the blue arrowed arc in Fig. 5) repeats until the CLOPA constraint is satisfied.

⑥ *Installation:* The finalized design then moves to the installation phase.

V. Integrated Design Example

In this section, we present an integrated design example for a process control system to illustrate the proposed CLOPA and integrated lifecycle. The system described in this section is a real testbed located in Qatar University and comprises the process simulator and the full plant control system. As the integrated design lifecycle is substantial, with some steps outside the scope of this article (e.g., SIS architectural design and security risk assessment), it is not possible to present the design process in full details. However, we try to focus on the big picture as related to the proposed CLOPA, while discussing briefly each design step. Wherever needed, we refer the reader to relevant references for further details.

A. CPS Description

We consider the continuous stirred tank reactor (CSTR) process illustrated in Fig. 6. The reactor vessel has an inlet stream carrying the reactant A, an outlet stream carrying the product

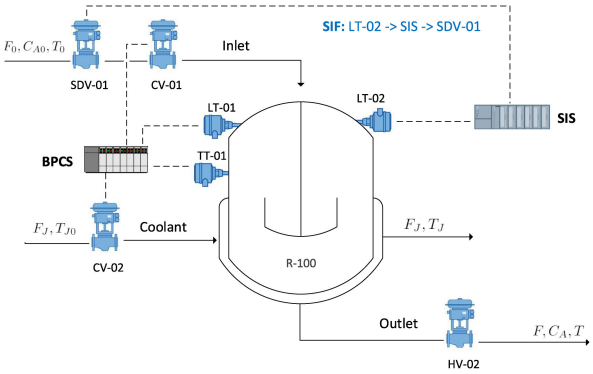

Fig. 6. Reactor piping and instrumentation diagram. ISA standard symbols are not strictly followed for illustration purposes.

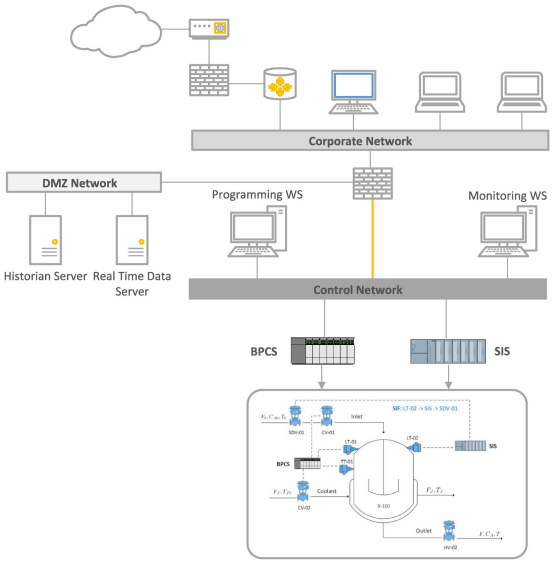

Fig. 7. CPS architecture for an industrial control system testbed, following NIST 800-82 guidelines. The firewall blocks any direct communication between the corporate network and the control network. Plant floor information is accessible only via the DMZ.

B, and a cooling stream carrying the cooling fluid into the surrounding jacket to absorb the heat of the exothermic reaction. A first-order reaction takes place where a mole fraction of reactant A is consumed to produce product B. The process has a level control loop (LT-01 $\rightarrow$ BPCS $\rightarrow$ CV-01) to maintain the liquid level in the reactor, and a temperature control loop (TT-01 $\rightarrow$ BPCS $\rightarrow$ CV-02) to control the reaction rate. The SIF (LT-02 $\rightarrow$ SIS $\rightarrow$ SDV-01) protects the reactor from the overflow hazard and will be explained later in this section. For more detailed explanation about the process including the state space model, the reader is referred to [12].

The CSTR process is controlled by the industrial control system shown in Fig. 7, which follows NIST 800-82 standard with one firewall and a DeMilitariZed (DMZ) zone [13]. The corporate network cannot communicate directly with the control network. The only allowable information flow paths via the firewall are from the control network to the data logging servers in the DMZ zone and from the corporate network to the DMZ for information retrieval. The BPCS and the SIS have Modbus/TCP communication over the control network [6].

B. Integrated Lifecycle

In the following discussion, we follow the integrated lifecycle in Fig. 5, and as per the itemized steps in Section IV-B, we have the following.

① *SIS safety lifecycle—HAZOP:* Table II shows the HAZOP sheet for the CSTR process. Each row contains: 1) the possible hazard; 2) all possible initiating events for each hazard whether mechanical or electronic failures; 3) consequences if the hazard occurred, including safety, financial, and environmental losses; 4) existing safeguards that could prevent the hazard from propagating and causing the consequences; and 5) the risk rank, which is typically a function of the consequences. There are two identified hazards for the reactor process: high level causing an overflow hazard, and high temperature that may lead to reactor runaway and possible meltdown. Both the hazards have high and very high risk rankings; therefore, the two risk scenarios qualify for further LOPA assessment. In the following, we limit our discussion to the high-level hazard scenario only. High-temperature hazard could be treated similarly.

② *BPCS cyber security lifecycle:* We need to calculate $P[A_B]$ and $P[A_{SB}]$ for the BPCS, the probability that the BPCS fails due to a direct attack and a SIS-pivot attack, respectively, in a way that generates the high-level process hazard. We conducted vulnerability identification on the CPS network in Fig. 7, constructed the attack trees, and carried out penetration testing to verify the vulnerability findings. We assumed an attacker profile where attacks are selected randomly. The total number of semantically relevant attacks is found to be $|\mathcal{A}_r| = 42$. We assume that relevant attacks represent 10^{-4} of all possible attacks, i.e., $|\mathcal{A}_r|/|\mathcal{A}| = 10^{-4}$. Therefore, according to Lemma 3.1, we obtain an initiating event likelihood $10^{-4}\lambda$ and a weight factor $\gamma_a = 0.024$ for each attack a at the leaf nodes of the attack tree. As the full details of vulnerability analysis, attack design, and penetration testing are beyond the scope of this article, we refer the interested reader to [10] and [14].

To compromise the BPCS, we assume the more realistic situation with no insider threat and no direct communication from the corporate network to the control network. In this scenario, the attacker has to detour to compromise the real-time (RT) server in the DMZ and use it as a pivot to attack the BPCS, either directly or via the monitoring workstation (designated HMI hereafter) that has legitimate communication with the BPCS. We start with the assumption that one of the corporate network PCs that has legitimate access to the RT server is compromised. There are several well-known attack vectors in the IT security domain to achieve such compromise, such as a spam email, a web service vulnerability, or an external malware USB, just to name a few. Fig. 8 shows an abstract attack tree that summarizes the BPCS compromise paths, where the database server compromise is a prerequisite attack step. In the following, we expand each of the leaf nodes in this abstract attack tree into the corresponding detailed attack trees. More detailed treatment of each attack tree as well as penetration testing could be found in [10].

TABLE II
PARTIAL HAZOP SHEET FOR THE REACTOR PROCESS

Hazard	**Initiating Event** (Cause)	**Consequences**	**Safeguards** (IPL)	Risk Rank
High Level (Reactor overflow)	BPCS failure OR Human error (misaligned valves)	2 or more fatalities (safety), Product loss (financial), Environmental contamination (environment)	Reactor dike (Mitigation)	High
High Temperature (Reactor Meltdown/explosion)	BPCS failure OR Coolant inlet control valve fully (partially) closed OR Inlet valve stuck fully open	10 or more fatalities (safety), Product loss (financial), Environmental contamination (environment)	None	V. High

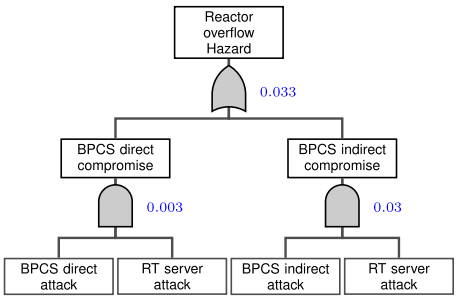

Fig. 8. Abstract attack tree to compromise the BPCS to generate overflow process hazard for the CSTR reactor. Leaf nodes are further expanded in Figs. 9–11.

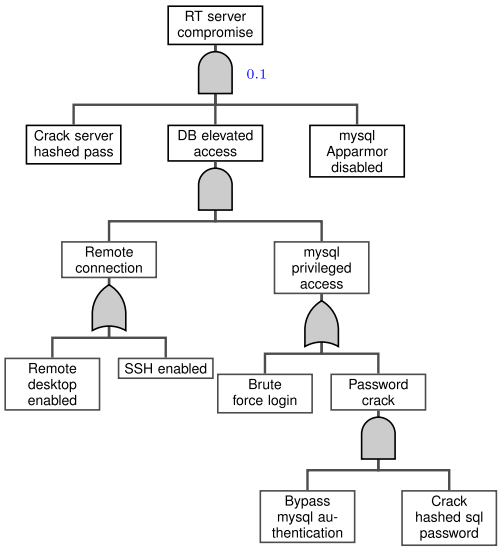

Fig. 9. RT server attack tree. Database vulnerabilities are exploited to gain root access and use the RT server as a pivot to attack the control network. DB elevated access is a prerequisite to crack the server hashed password. Leaf nodes that represent a distinct attack have a weight factor $\gamma_a = 0.024$ that is combined with their success probability. Therefore, Bypass mysql and Crack hashed sql password, Brute force login, SSH enabled, remote desktop enabled, and each has a weight factor combined with their success probability.

Fig. 9 shows the RT server attack tree. The basic idea is to exploit mysql database vulnerabilities via SSH to obtain the Linux server password Hash Dump and possibly crack the password to achieve privilege escalation and gain full control over the RT server. The probability of success of such an attack depends on several factors, including mysql configuration settings to allow brute-force login attack, mysql login password strength, configuration of mysql security monitoring app, and whether SSH is enabled. For the purpose of this case study, we choose this probability arbitrarily as 0.1. It should be highlighted that the attack tree does not have the sequence semantics to represent a sequence of attack steps. For example, the remote connection step has to be executed before the mysql privileged access in Fig. 9. We represent this sequence by the AND gate aggregator, noting that in some other cases the AND gate may represent simultaneous attack steps. For more information on attack trees and their semantics, the reader is referred to [15].

Fig. 10 shows the BPCS attack tree, which is divided into two main parts: DoS attack and integrity attack. The DoS attack may not lead to a reactor overflow unless there is a concurrent process disturbance that could not be controlled with the DoS-induced delayed BPCS control response. The probability of such disturbance could be estimated from plant information. The integrity attack injects a low-level measurement value for LT-01 to drive the BPCS controller to increase valve CV-01 opening or directly forces control valve CV-01 to open 100%. This will cause a reactor overflow if the SIS is not activated. The injection of the malicious value in the control loop could be accomplished by either gaining access to the controller and overwriting the control program or more simply sending Modbus packets to the controller with the malicious values. Modbus attack is much easier to launch but requires configuration data to identify the Modbus register address for either LT-01 or CV-01. The probability of BPCS indirect attack is chosen arbitrarily as 0.03.

Finally, Fig. 11 shows the attack tree for BPCS attack via the HMI. The attack is launched by remote desktop connection to the HMI and legitimately controlling CV-01 via the GUI. This indirect attack is easier than targeting the BPCS directly as it does not require knowledge about the controller configuration or Modbus register addresses associated with the sensor and valve of the targeted control loop. This is because all the information is already programmed in the GUI software. The probability of BPCS indirect attack is estimated to be 0.3. Using Fig. 8 and the three presented attack trees in Figs. 9–11, the total probability of BPCS attack that leads to an overflow hazard could be estimated by $P[A_B] \approx 0.033$.

It should be highlighted that the assignment of a probability measure to the success of attack actions is subject to debate in the research community, and there is no published agreed-upon data as in the case of reliability failure data. One approach is to use attack databases, such as NIST National Vulnerability Database (NVD) [16], to estimate the probability of a cyber attack success based on attributes such as required knowledge level and attack

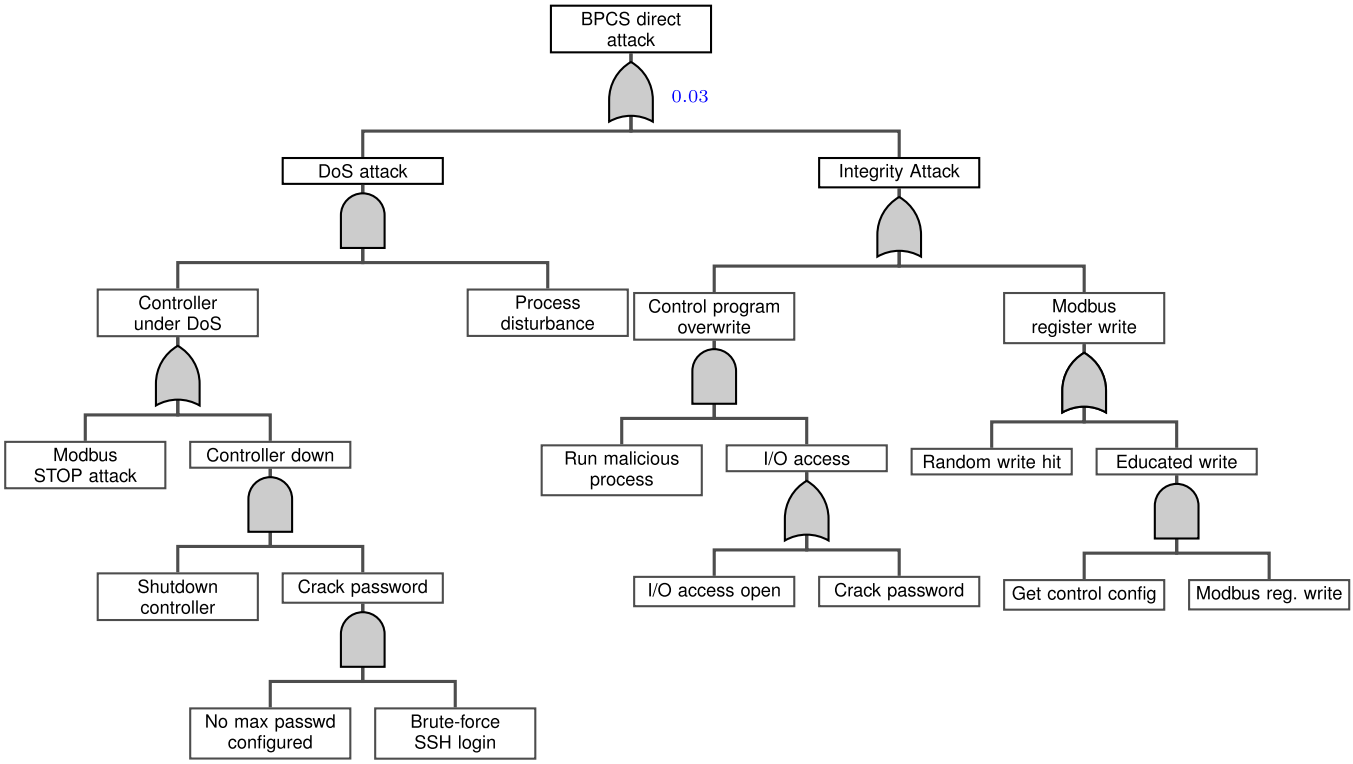

Fig. 10. Attack tree for BPCS compromise to generate a reactor overflow hazard. A DoS attack synchronized with a process disturbance or an especially crafted integrity attack would cause the CSTR to overflow. Leaf nodes that represent a distinct attack have a weight factor $\gamma_a = 0.024$ that is combined with their success probability.

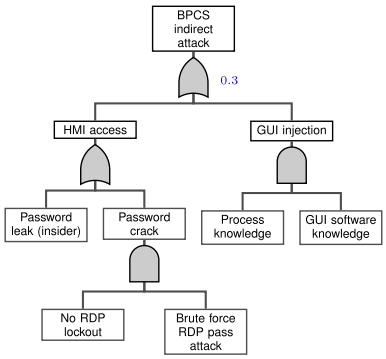

Fig. 11. HMI-BPCS indirect attack tree. The compromised HMI is used to embed the attack against the BPCS using the legitimate traffic between the GUI and the BPCS control program. Leaf nodes that represent a distinct attack have a weight factor $\gamma_a = 0.024$ that is combined with their success probability.

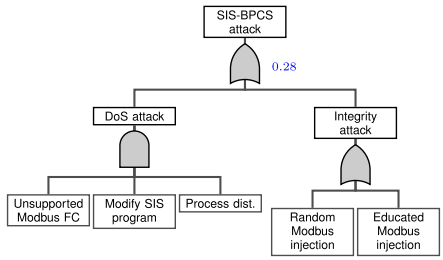

Fig. 12. SIS-BPCS attack tree. The SIS is used as a pivot to launch either a DoS attack or a crafted integrity attack that results in reactor overflow. Leaf nodes that represent a distinct attack have a weight factor $\gamma_a = 0.024$ that is combined with their success probability.

difficulty. However, this approach has the drawback that it does not take into account the specifics of each organization. In this article, we rely on the experience obtained during the penetration testing carried out by the research team in combination with NVD to assign the probability measures. This does not impact the analysis as the presented case study is meant for illustration purposes to explain the design process.

To calculate the probability of BPCS cyber attack leading to a process hazard given an SIS cyber compromise $P[A_{SB}]$, we focus on Modbus attack vectors for both integrity and DoS attacks. Integrity attacks target sensor LT-01 or valve CV-01 as before, either randomly or using leaked Modbus register configuration. DoS attack could be launched by utilizing non-programmed Modbus function code hoping that it would crash the BPCS Modbus master. Fig. 12 summarizes the attack tree, and the probability is chosen arbitrarily as $P[A_{SB}] \approx 0.2813$. To summarize, the desired outcome from the BPCS security lifecycle is $(P[A_B], P[A_{SB}]) = (0.033, 0.2813)$.

It should be noted that complete attack trees for the given BPCS and CPS architecture could span multiple pages. However, full attack trees may obscure the analysis and will serve no additional insight. Therefore, the simplified attack trees presented here act as a better illustration of the design methodology. For more in-depth treatment of the cyber risk assessment for the presented case study, refer to [10].

TABLE III
LOPA SHEET FOR THE CSTR OVERFLOW HAZARDOUS SCENARIO

Initiating Event	Likelihood λ_i (/yr)	Tank Dike	Safety Procedure	Human Intervention	BPCS ($P[B_p]$)	TMEL
Inlet flow surge	10^{-1}	10^{-2}	1	10^{-1}	10^{-1}	10^{-6}
Downstream flow blockage	10^{-1}	10^{-2}	10^{-1}	10^{-1}	10^{-1}	10^{-6}
Manual valves misalignment	10^{-1}	10^{-2}	10^{-1}	10^{-1}	10^{-1}	10^{-6}
BPCS physical Failure	$10^{-1}(\lambda_b)$	10^{-2}	1	10^{-1}	1	10^{-6}
BPCS attack Failure	$10^{-2}(\lambda_c)$	10^{-2}	1	10^{-1}	1	10^{-6}

Numbers in each cell represent the probability of failure of the associated protection layer.

TABLE IV
CSTR CLOPA—CALCULATED PARAMETER VALUES

LOPA Parameter	Value	Source
$\Sigma_{i=1}^{3}\lambda_i$	0.3	LOPA Sheet
$P[\mathcal{L}]$	0.001	LOPA Sheet
λ_b	0.01	LOPA Sheet
λ_c	0.01	LOPA Sheet
$P[B_p]$	0.01	LOPA Sheet
α_1	1.13×10^{-4}	CLOPA Parameter-Calculated Eq. (10)
α_2	1.17×10^{-4}	CLOPA Parameter-Calculated Eq. (11)
β	10^{-6}	CLOPA Parameter-Calculated Eq. (12)
γ_1	1.4868×10^{-4}	CLOPA Parameter-Calculated Eq. (14)
γ_2	7.5785×10^{-6}	CLOPA Parameter-Calculated Eq. (15)
γ_3	1.1686×10^{-4}	CLOPA Parameter-Calculated Eq. (16)

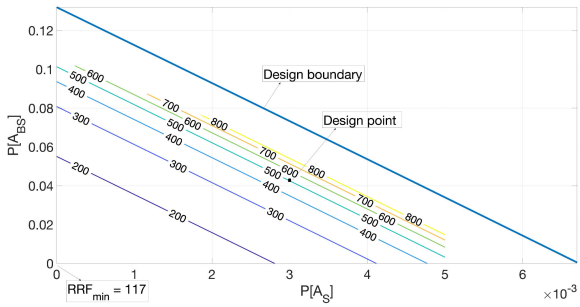

Fig. 13. CSTR case study: CLOPA design region with the contour plot for the RRF.

③ *SIS safety lifecycle—CLOPA:* Table III shows the LOPA sheet for the CSTR overflow hazard identified from the HAZOP, where the initiating event likelihoods and failure probabilities are adopted from [17] and [18]. The BPCS cyber attack likelihood is calculated as $\lambda_c = 10^{-4}\lambda = 0.01$ per year, assuming $\lambda = 100$ per year. Note that human intervention is considered a protection layer assuming that there is sufficient time for the operation team to manually isolate the reactor in the field. Some conservative approaches omit any human intervention or safety procedure from the LOPA.

From the LOPA sheet, we extract the event likelihood values to calculate the CLOPA model parameters using (10)–(12) and (14)–(16), along with $(P[A_B], P[A_{SB}]) = (0.033, 0.2813)$ from the BPCS security lifecycle. Table IV summarizes the parameter values. Substituting into the CLOPA constraint (13), we obtain

$$P[S_p] \leq \frac{1 - 148.68P[A_S] - 7.6P[A_{BS}](1 - P[A_S])}{117(1 - P[A_S]) - 7.6P[A_{BS}](1 - P[A_S])}. \quad (27)$$

Our objective now is to design an SIF with architecture $\mathcal{A}$ that satisfies (27) in order to achieve the required process safety objective as defined by the TMEL in the LOPA analysis. Our initial design for the SIF will comprise a level sensor (LT-02), a logic solver (SIS), and a shutdown valve (SDV-01), as illustrated in Fig. 6. The SIF will take an independent action upon reactor overflow and will close the inlet shutdown valve. The architecture of the SIF could vary through design iterations to achieve the required safety. As an example, sensors may be duplicated or sometimes triplicated to achieve higher reliability, and the SIS architecture may include redundant CPU modules. We note that for a perfectly secured SIS ($P[A_S] = P[A_{BS}] = 0$), $P[S_p] \leq 1/117$, or equivalently RRF ≥ 117. This is the minimum achievable RRF. Since for practical systems there is no zero probability of cyber security attack failures, our SIS design is expected to have an RRF > 117.

Using the calculated LOPA parameter values, the design region (18) and boundary (19) are defined by

$$P[A_{BS}] \leq 0.132\left(\frac{1 - 148.68P[A_S]}{1 - P[A_S]}\right) \quad (28)$$

where the design boundary is defined when the equality holds. The contour lines for the RRF in (20) are defined by

$$P[A_{BS}] = \left(\frac{15.42}{C - 1}\right)\left(\frac{(C - 0.008) - (C - 1.27)P[A_S]}{1 - P[A_S]}\right) \quad (29)$$

for different values C of the RRF. The design region and the contour lines are plotted in Fig. 13. We note that as we approach the design boundary, either by increasing $P[A_S]$ or $P[A_{BS}]$, the RRF rapidly increases such that it is not possible to plot the contour lines in this region in a visible way. The design in this region is very sensitive to input variations (i.e., a very small variation in probabilities will result in a very large change in RRF). Therefore, the design point should be selected as far as possible from the design boundary. To further illustrate the increase in RRF, Fig. 14 shows a 3-D plot for the RRF as it varies with both $P[A_S]$ and $P[A_{BS}]$. It should be evident from the 3-D plot that for small values of $P[A_S]$, the function gradient is smaller, resulting in a less-sensitive design to probability variations. At larger values of $P[A_S]$ near the design boundary, the RRF increases exponentially with $P[A_{BS}]$. These results could be verified by calculating the gradient of (13).

To proceed with the design process, we pick the point $P[A_S] = 0.003$ as a reasonable probability value for SIS direct attack failure that is away from the steepest ascent region in Fig. 14. We now need to choose a practical value of $P[A_{BS}]$

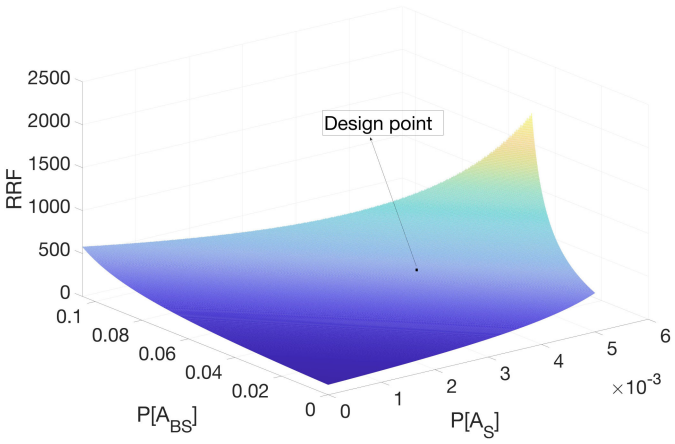

Fig. 14. CLOPA RRF as it varies with SIS security failure probabilities. Steepest ascent region to the right should be avoided when selecting the operating point.

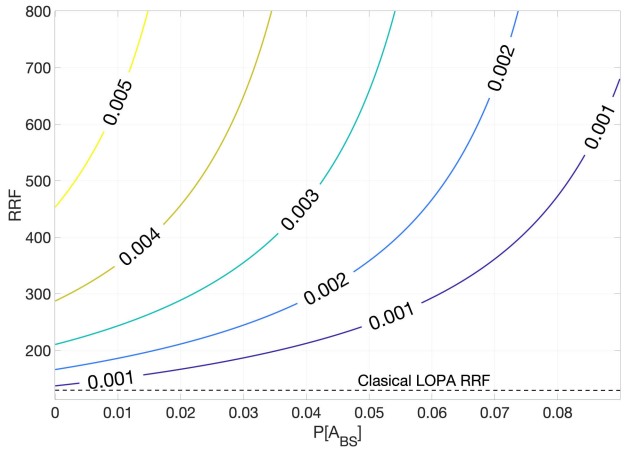

Fig. 15. Increase of the RRF with $P[A_{BS}]$. Each curve corresponds to the fixed value indicated for $P[A_S]$.

that results in an achievable target RRF. With the help of Fig. 13 and contour lines, $P[A_S] = 0.003$ intersects the contour line for RRF $= 500$ at $P[A_{BS}] = 0.0426$. Alternatively, the value of $P[A_{BS}]$ could be obtained from (29) by setting $C = 500$ and $P[A_S] = 0.003$. The design point (0.003, 0.0426, 500) is indicated in Figs. 13 and 14. The design and verification of the SIF then resume according to IEC 61511 to develop an architecture $\mathcal{A}$ that satisfies the combined CLOPA requirement: RRF ≥ 500, $P[A_S] \leq 0.003$, and $P[A_{BS}] \leq 0.0426$. The detailed design and verification of the SIF are outside the scope of this article (refer to [4] for more details). To complete the case study, we will assume that the design engineer came up with an architecture $\mathcal{A}$ that was verified using vendor data, resulting in reliability $\text{RRF}_v = 600$, with a design margin from the required RRF $= 500$.

④ *SIS cyber security lifecycle:* The resulting SIF architecture $\mathcal{A}$ is used to carry out the SIS security lifecycle, similar to the BPCS security risk assessment in step 2 of the design process. As the SIS detailed design and verification are not in the scope of this article, we will assume for the sake of illustration that the architecture $\mathcal{A}$ results in a cyber system configuration that has a higher probability of SIS security attack failure $P'[A_S] = 0.004$ while reducing the BPCS pivot attack failure probability to $P'[A_{BS}] = 0.03$ via securing the BPCS–SIS link.

⑤ *Safety lifecycle—CLOPA:* The architecture $\mathcal{A}$ results in $P'[A_S] = 0.004$, $P'[A_{BS}] = 0.03$, and $\text{RRF}_v = 600$. We need to verify if these values satisfy the CLOPA constraint (27). Plugging the probability values results in $P[S_p] \leq 1.54 \times 10^{-3}$, or equivalently RRF ≥ 643. As $\text{RRF}_v = 600 < 643$, the architecture has to be modified either by reducing further the cyber attack failure probabilities or by increasing the system reliability via fault tolerance techniques. It may take the design engineer multiple iterations until the design achieves the CLOPA constraint. In practice, the iterations do not involve a complete architectural redesign, but rather changing the redundancy scheme or security hardening in order to achieve the design objective. To conclude the case study example, we will assume that the design engineer came up with an architecture that preserves the aforementioned probability values while increasing the RRF to 650. This concludes the design process and the system moves to the implementation phase. The case study design values are superimposed on the iterative design process in Fig. 4 as an illustration. ■

C. Classical LOPA Error

Classical LOPA ignores cyber attack probabilities altogether. For the given problem, it results in RRF $= 113$ as per (17). The minimum CLOPA RRF occurs for a perfectly secured safety system where $P[A_S] = P[A_{BS}] = 0$, achieving RRF $= 117$. Therefore, the minimum error between LOPA and CLOPA RRF estimation is 4. The error gets worse as security failure probabilities increase. For the given design point $P[A_S], P[A_{BS}] = (0.003, 0.0426)$, the classical LOPA error is $e_{\text{RRF}} = 378$. This is a significant amount of error that results in the design of a less reliable system that will not achieve the target risk level. Fig. 15 better illustrates the error increase with increasing the security failure probability $P[A_{BS}]$ for different values of $P[A_S]$. For small values of $P[A_S]$, the curves show slow increase in the RRF with $P[A_{BS}]$. As $P[A_S]$ increases, the RRF increase becomes exponential. A similar contour figure for fixed $P[A_{BS}]$ values could be generated. The design point for the case study $P[A_S] = 0.003$ was chosen as a tradeoff between an achievable cyber attack probability value and a moderate rate of increase for the RRF. The 3-D plot for the error in RRF versus $P[A_S], P[A_{BS}]$ is identical to Fig. 14, except by shifting down the 3-D curve by 113, the LOPA RRF value; therefore, it is omitted to avoid repetition.

D. Sensitivity Analysis

Calculating the probability of a security failure is a debatable subject in the research community, especially with the lack of statistical data that are available for physical failures. One question that comes to mind is the robustness of the developed CLOPA model to probability variations. We conducted a numerical analysis to calculate the partial derivatives of the RRF with respect to $P[A_S], P[A_{BS}]$. The two partial derivative plots are very similar to Fig. 14 and omitted for space limitation. For

small probability values, the change in the RRF is in the range of 15% for 10^{-3} change in $P[A_S]$. As probabilities increase and we approach the decision boundary, the change in the RRF jumps to around 80% for 10^{-3} change and increases exponentially as we get closer to the decision boundary. A similar behavior is exhibited with $P[A_{BS}]$ change (figure omitted for brevity). However, the change in the RRF has much lower percentage, ranging from 7% for small probability values, and increasing to around 37% as we approach the decision boundary. We highlight the following three key observations.

1) For small cyber failure probability values, the model sensitivity is acceptable since the SIL levels have an order of magnitude ratio, so a small percentage change would likely keep the system requirement in the same SIL category. However, this requires that the probability error is in the range of 10^{-3}.
2) The model is more sensitive to direct attack failure probabilities than BPCS pivot attacks.
3) We should always try to design our system as far as possible from the decision boundary. The model sensitivity with respect to probability changes increases as we approach the decision boundary.

VI. Related Work

HAZOP has been the dominant risk assessment method for the process industry for over 30 years [5], [19], [20]. LOPA has been used in conjunction with HAZOP to design SISs and specify the SIL for each SIF [21]. Because of the wide adoption of LOPA by industry due to its systematic approach and quantitative risk assessment capability, LOPA has been included as one of the methods in IEC 61511-3 standard with several illustrating examples [4]. The LOPA approach has been applied to physical security risk analysis in [22]. However, to the best of authors' knowledge, there is no research work on integrating security attacks in the LOPA framework for SIS design.

There are emergent standardization initiatives to address safety and security coordination in CPSs. IEC 62443-4-1 (Security for Industrial Automation and Control Systems—Part 4-1: Secure Product Development Lifecycle Requirements) is a standard developed by the ISA-99 committee with the purpose to extend the existing safety lifecycle at different phases to include security aspects to ensure safe CPS design [23]. IEC TC65 AHG1 is a recently formed group linked to the same technical committee developing IEC 61508 and IEC 62443 to consider how to bridge functional safety and cyber security for industrial automation systems [24]. IEC 62859 (Nuclear Power Plants—Instrumentation and Control Systems—Requirements for Coordinating Safety and Cyber Security) is a standard derived from IEC 62645 for the nuclear power industry to coordinate the design and operation efforts with respect to safety and cyber security [25]. DO-326 (Airworthiness Security Process Specification) is a standard for the avionics industry that augments existing guidelines for aircraft certification to include the threat of intentional unauthorized electronic interaction to aircraft safety [26]. A taxonomy of dependable and secure computing is introduced in [27] in order to facilitate the communication among different research communities. The concepts and taxonomy presented are a result of a joint committee on "Fundamental Concepts and Terminology" that was formed by the Technical Committee on Fault-Tolerant Computing of the IEEE Computer Society and the IFIP WG 10.4 "Dependable Computing and Fault Tolerance." A preliminary work on the research in this article that combines the two research directions stated below is presented in [28].

A. Lifecycle Integration

Kornecki and Liu [29] use fault tree analysis to combine both safety and security failures in one unified risk assessment framework for the aviation industry. The outcome of the risk assessment is used to define both safety and security requirements. A road map for cyber safety engineering to increase air traffic management system resilience against cyber attacks is proposed in [30]. The V-shaped model to develop embedded software for CPS is augmented with security actions in [31]. The integration of IEC 61508 safety standard and IEC 15408 for IT security is described in [32]–[34] for building automation systems. Sørby [35] describes in more detail the integration of IEC 61508 safety lifecycle and the CORAS approach to identify security risks [36]. An approach to align safety and security during the different stages of system development lifecycle is proposed in [37]. The approach, called Lifecycle Attribute Alignment, ensures compatibility between safety and security controls developed and maintained during the system development lifecycle. HAZOP, a predominantly used method for safety risk assessment in the process industry, is modified in [38] to include security failures. The authors introduce new guide words, attributes, and modifiers for security components akin to traditional HAZOP limited to safety failures. Failure Mode and Effect Analysis is extended in [39] to include security vulnerabilities, suggesting the name Failure Mode Vulnerability and Effect Analysis. For a survey on the integration of safety and security in the CPS, refer to [2].

B. Model-Based Risk Assessment

Several graphical methods have been used to combine safety and security analysis. Goal structuring notation (GSN) is a graphical notation used to model requirements, goals, claims, and evidence of safety arguments [40]. The SafSec research project for the avionics industry elaborates on the use of GSN to integrate both safety and security arguments in one representation [41]. A similar approach is used in [42], where the authors apply the nonfunctional requirement (NFR) approach to quantitatively assess the safety and security properties of an oil pipeline CPS. NFR is a technique that allows simultaneous safety and security graphical representation and evaluation at the architectural level.

The simplicity and wide adoption of fault and attack trees promoted the research work to merge both modeling tools. The integration of fault trees and attack trees is considered in [43] in order to extend traditional risk analysis to include cyber attack risks. A quantitative analysis is proposed by assigning probabilities to tree events. Similarly, fault tree analysis is used in [29] to analyze safety/security risks in aviation software. Steiner and Liggesmeyer [44] extend component fault trees to

contain both safety and security events. Both the qualitative and quantitative analyses are performed to assess the overall risk. The quantitative analysis is enabled by assigning probabilities to safety events and categorical rating (low, medium, and high) for security events. Kumar and Stoelinga [45] translate the combined fault-attack tree into stochastic time automata to enable quantitative risk analysis. The use of bow-tie diagrams and analysis in place of fault trees is reported in [46], where it is integrated with attack trees for combined safety–security risk assessment.

Given the limited semantics of fault trees, Boolean-logic-driven Markov process (BDMP) graphical formalism introduced in [47] has been used to integrate safety and security events. The approach integrates fault trees with the Markov process at the leaf node level and associates a mean time to success for security events and a mean time to failure for safety events. This allows both a qualitative and a quantitative risk assessment for the given system. The formalism also enables the modeling of detection and response mechanisms without a need for model change. The work in [48] applies BDMP formalism to a pipeline case study, illustrating different types of safety–security interdependencies. In [49], Stuxnet attack is modeled using the BDMP and a quantitative risk analysis is carried out on the industrial control system.

Petri nets have also been proposed to overcome the limitations of fault trees. A formalism for safety analysis named state/event fault trees is reported in [50]. In this formalism, both deterministic state machines and Markov chains are combined, while keeping the visualization of causal chains known from fault trees. This formalism is extended in [51] to include an attacker model to deal with both safety and security. Similarly, stochastic Petri nets have been used in [52] to model the impact of intrusion detection and response on CPS reliability and in [53] to assess the vulnerabilities in supervisory control and data acquisition systems. Bayesian belief networks are also considered as one of the model-based approaches. In [54], a Bayesian belief network is used to assess the combined safety and security risk for an oil pipeline example.

The Unified Modeling Language (UML) commonly used in software engineering has also been used for safety and security risk assessment. Misuse cases for UML diagrams have been used to define safety requirements in [55] and security requirements in [56], independently. A combined process for Harm Assessment of Safety and Security has been proposed in [57] based on both UML and HAZOP studies. UMLsafe [58] and UMLsec [59] are two UML extensions that enable modeling of safety and security requirements, respectively. The combined UMLsafe/UMLsec is proposed in [60] for safety–security codevelopment. SysML-sec, a SysML-based model driven engineering environment, is used in [61] for the formal verification of safety and security properties.

System-theoretic process analysis (STPA) was developed as a new hazard analysis technique to evaluate the safety of a system [62]. Friedberg *et al.* [63] extend the STPA to include system security aspects in the analysis. The expanded approach is named STPA-SafeSec and demonstrated on a use case in the power grid domain. The system-theoretical accident model and process (STAMP) is applied to the Stuxnet attack in [64], showing that the attack could have been avoided if the STAMP was applied during design time.

VII. Conclusion

Classical safety assessment methods do not take into account failures due to cyber attacks. In this article, we showed quantitatively that overlooking security failures could bias the risk assessment, resulting in underdesigned protective systems. In addition, the design of safety and security subsystems for complex engineering systems cannot be carried out independently, given their strong coupling as demonstrated in this article. Although the design becomes more complicated when considering cyber attacks, the development of new software tools or the modification of existing industrial tools could automate the process.

In this article, we considered the control system (BPCS) design as given, following common industrial practice. The joint optimization of both BPCS and SIS designs, from both safety and security perspectives, is a potential extension for the presented work. In addition, the presented integrated lifecycle relies in part on designer's experience to make design decisions to achieve the system requirements. Optimal system design that captures possible safety and security design choices with associated financial cost could provide a better quantitative approach to find the optimal system operating point rather than relying on design heuristics. Furthermore, the integration of both the safety and security lifecycles into model-based design toolchains is crucial for adoption by industry.

Finally, the work presented in this article discusses the impact of cyber security failure on system safety. A closely related problem is how safety failures could impact cyber security. There is not much work in this direction, perhaps because the focus in CPSs is always on safety, considering the security of the cyber system as a secondary issue. Nevertheless, this is an important problem. On the one hand, a simple safety failure may be injected to cause a security compromise that may be exploited to produce a higher security compromise that could lead to a greater safety hazard. On the other hand, both directions, i.e., Safety $\rightarrow$ Security and Security $\rightarrow$ Safety, are closely related and interacting, and therefore, optimizing a CPS performance with respect to safety/security or both cannot be fully achieved without understanding the two types of interactions.

VIII. Source Code

The source code for the CLOPA in the form of MATLAB m files to regenerate the research results including the case study is located at https://github.com/Ashraf-Tantawy/CLOPA.git.

Appendix A

Proof of Lemma 3.1

Proof: The aggregate likelihood of all attacks to cause a hazard taking into account BPCS and SIS protection could be approximated by (neglecting higher order probability terms)

$$\Lambda = \lambda \sum_{a \in \mathcal{A}_r} \alpha_a P_a[S, B]. \tag{30}$$

TABLE A1
CLOPA MODEL PARAMETERS

Symbol	Description	Type	Calculation Method/ Data Source
λ_i	Initiating event i likelihood (/yr)	Parameter	Reliability data
λ_p	BPCS physical failure event likelihood (/yr)	Parameter	Reliability data
λ_a	BPCS cyber attack a likelihood (/yr)	Parameter	Refer to Section III-B
λ_c	BPCS semantically-related attacks likelihood (/yr)	Parameter	Refer to Section III-B, Lemma III.1
λ	BPCS cyber attack likelihood for all attacks	Parameter	Statistical attack data
α_a	Probability of selecting attack a by the attacker	Parameter	Attacker profile model
γ_a	Weight factor for the attacks for the equivalent BPCS	Parameter	Refer to Lemma III.1
$P[\mathcal{L}_i]$	Probability of failure of all protection layers for initiating event i	Parameter	Reliability data
TMEL	Target Mitigated Event Likelihood	Parameter	Determined by the corporate policy
$P[B_c]$	Probability of BPCS security failure	Intermediate design variable	BPCS security risk assessment
$P[B_p]$	Probability of BPCS physical failure	Parameter	Reliability data
$P[A_B]$	Probability of BPCS direct security failure	Parameter	BPCS security risk assessment
$P[A_{SB}]$	Probability of BPCS SIS-pivot security failure	Parameter	BPCS security risk assessment
$P[S_c]$	Probability of SIS security failure	Intermediate design variable	SIS security risk assessment
$P[S_p]$	Probability of SIS physical failure	Design variable	SIS security risk assessment
$P[A_S]$	Probability of SIS direct security failure	Design variable	SIS security risk assessment
$P[A_{BS}]$	Probability of SIS BPCS-pivot security failure	Design variable	SIS security risk assessment
$P[S_c, B_c]$	Probability of simultaneous SIS and BPCS security failure	Intermediate design variable	BPCS & SIS security risk assessment
$\alpha_1 - \alpha_2$	-	Auxiliary parameters	Eq. (10), (11)
$\gamma_1 - \gamma_3$	-	Auxiliary parameters	Eq. (14) to (16)
$\zeta_1 - \zeta_3$	-	Auxiliary parameters	Eq. (23) to (25)
β	-	Auxiliary parameters	Eq. (12)

Variables designated as "Design variable" are with respect to CLOPA, but could be a design variable of another assessment, such as $P[A_B]$, derived from BPCS security risk assessment. Variables designated as "intermediate design variables" could be expressed in terms of design variables.

Using (8) to expand the joint probability, we obtain

$$\Lambda = \lambda \sum_{a \in \mathcal{A}_r} \alpha_a \left(\eta_1 + \eta_2 P[B_c^a] + \eta_3 P[S_c] P[B_c^a | S_c]\right) \quad (31)$$

where $P[B_c^a]$ represents the probability of BPCS security failure with respect to attack a, and η_1, η_2, and η_3 are probability terms not dependent on the attack a. Expanding, we obtain

$$\Lambda = \lambda \left(\sum_{a \in \mathcal{A}_r} \alpha_a\right) \times \quad (32)$$

$$\left(\eta_1 + \eta_2 \sum_{a \in \mathcal{A}_r} \gamma_a P[B_c^a] + \eta_3 \sum_{a \in \mathcal{A}_r} \gamma_a P[S_c] P[B_c^a | S_c]\right) \quad (33)$$

where $\gamma_a = \alpha_a / \sum_{a \in \mathcal{A}_r} \alpha_a$. Ignoring higher order probabilities, we obtain

$$\Lambda \approx \lambda \left(\sum_{a \in \mathcal{A}_r} \alpha_a\right) \times \quad (34)$$

$$\left(\eta_1 + \eta_2 P\left[\sum_{a \in \mathcal{A}_r} \gamma_a B_c^a\right] + \eta_3 P\left[S_c, \sum_{a \in \mathcal{A}_r} \gamma_a B_c^a\right]\right). \quad (35)$$

Comparing (31) and (35), the second and third terms in (35) represent an equivalent BPCS with a combined attack vector $\mathcal{A}_r$, where each attack a is weighted by γ_a. In addition, the likelihood of this combined attack vector is $\lambda(\sum_{a \in \mathcal{A}_r} \alpha_a)$.

REFERENCES

[1] S. Kriaa, L. Pietre-Cambacedes, M. Bouissou, and Y. Halgand, "A survey of approaches combining safety and security for industrial control systems," *Rel. Eng. Syst. Saf.*, vol. 139, pp. 156–178, 2015.

[2] X. Lyu, Y. Ding, and S. H. Yang, "Safety and security risk assessment in cyber-physical systems," *IET Cyber- Phys. Syst.: Theory Appl.*, vol. 4, no. 3, pp. 221–232, 2019.

[3] P. P. Gruhn, *Safety Instrumented Systems: Design, Analysis, and Justification*. Research Triangle Park, NC, USA: Instrum. Soc. Amer., 2006.

[4] *Functional Safety—Safety Instrumented Systems for the Process Industry Sector—Part 1: Framework, Definitions, System, Hardware and Application Programming Requirements*, IEC Standard IEC 61511-1:2016, 2016.

[5] J. Dunjó, V. Fthenakis, J. A. Vílchez, and J. Arnaldos, "Hazard and operability (HAZOP) analysis. A literature review," *J. Hazardous Mater.*, vol. 173, no. 1–3, pp. 19–32, 2010.

[6] A. Swales, *Open Modbus/TCP Specification*, vol. 29. Rueil-Malmaison, France: Schneider Electric, 1999.

[7] I. N. Fovino, A. Carcano, T. De Lacheze Murel, A. Trombetta, and M. Masera, "Modbus/DNP3 state-based intrusion detection system," in *Proc. 24th IEEE Int. Conf. Adv. Inf. Netw. Appl.*, 2010, pp. 729–736.

[8] A. Z. Faza, S. Sedigh, and B. M. McMillin, "Reliability analysis for the advanced electric power grid: From cyber control and communication to physical manifestations of failure," in *Proc. Int. Conf. Comput. Saf., Rel., Secur.*, 2009, pp. 257–269.

[9] G. Stoneburner, A. Goguen, and A. Feringa, "Risk management guide for information technology systems," NIST Special Publication 800-30, 2002.

[10] A. Tantawy, S. Abdelwahed, A. Erradi, and K. Shaban, "Model-based risk assessment for cyber physical systems security," *Comput. Secur.*, vol. 962020, Art. no. 101864.

[11] A. P. Moore, R. J. Ellison, and R. C. Linger, "Attack modeling for information security and survivability," Softw. Eng. Inst., Pittsburgh, PA, USA, Tech. Rep. CMU/SEI-2001-TN-001, 2001.

[12] A. Tantawy, S. Abdelwahed, and Q. Chen, "Continuous stirred tank reactors: Modeling and simulation for CPS security assessment," in *Proc. 11th Int. Conf. Comput. Intell. Commun. Netw.*, 2019, pp. 117–123.

[13] K. Stouffer, J. Falco, and K. Scarfone, "Guide to industrial control systems (ICS) security," NIST Special Publication 800-82, 2011, p. 164.

[14] A. Tantawy, "Automated malware design for cyber physical systems," in *Proc. 9th Int. Symp. Digit. Forensics Secur.*, 2021, pp. 1–6.

[15] S. Mauw and M. Oostdijk, "Foundations of Attack Trees (ser. Lecture Notes in Computer Science), vol. 3935. Berlin, Germany: Springer, 2006, pp. 186–198.

[16] *NVD—Home*, NIST, Gaithersburg, MD, USA, 2016.

[17] INTEF and NTNU, *OREDA Offshore and Onshore Reliability Data Volume 1—Topside Equipment*. Bærum, Norway: DNV, 2015.

[18] *Guidelines for Initiating Events and Independent Protection Layers in Layer of Protection Analysis*. Center Chem. Process Saf., New York, NY, USA, 2015.

[19] F. Crawley and B. Tyler, *HAZOP: Guide to Best Practice*. Amsterdam, The Netherlands: Elsevier, 2015.

[20] B. Skelton, *Hazop and Hazan: Identifying and Assessing Process Industry Hazards*. Rugby, U.K.: IChemE, 1999.

[21] A. M. Dowell, "Layer of protection analysis and inherently safer processes," *Process Saf. Prog.*, vol. 18, no. 4, pp. 214–220, 1999.

[22] F. Garzia, M. Lombardi, M. Fargnoli, and S. Ramalingam, "PSA-LOPA—A novel method for physical security risk analysis based on layers of protection analysis," in *Proc. Int. Carnahan Conf. Secur. Technol.* 2018, pp. 1–5.

[23] *Security for Industrial Automation and Control Systems – Part 4-1: Secure Product Development Lifecycle Requirements*, IEC Standard IEC 62443-4-1, 2018.

[24] H. Kanamaru, "Bridging functional safety and cyber security of SIS/SCS," in *Proc. 56th Annu. Conf. Soc. Instrum. Control Eng. Jpn.*, 2017, pp. 279–284.

[25] *Nuclear Power Plants—Instrumentation and Control Systems—Requirements for Coordinating Safety and Cybersecurity*, IEC Standard IEC 62859:2016, 2016.

[26] C. Torens, "Safety versus security in aviation, comparing DO-178C with security standards," in *Proc. AIAA Scitech 2020 Forum*, 2020, Art. no. AIAA 2020-0242.

[27] A. Avižienis, J. C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Trans. Dependable Secure Comput.*, vol. 1, no. 1, pp. 11–33, Jan.–Mar. 2004.

[28] A. Tantawy, A. Erradi, and S. Abdelwahed, "A modified layer of protection analysis for cyber-physical systems security," in *Proc. 4th Int. Conf. Syst. Rel. Saf.*, Rome, Italy, 2019, pp. 94–101.

[29] A. J. Kornecki and M. Liu, "Fault tree analysis for safety/security verification in aviation software," *Electronics*, vol. 2, pp. 41–56, 2013.

[30] C. W. Johnson, "CyberSafety: On the interactions between cybersecurity and the software engineering of safety-critical systems," *Lab. Med.*, vol. 21, no. 7, pp. 411–413, 2012.

[31] A. J. Kornecki and J. Zalewski, "Safety and security in industrial control," in *Proc. Annu. Workshop Cyber Secur. Inf. Intell. Res.*, 2010, Art. no. 77.

[32] T. Novak, A. Treytl, and P. Palensky, "Common approach to functional safety and system security in building automation and control systems," in *Proc. IEEE Int. Conf. Emerg. Technol. Factory Autom.*, 2007, pp. 1141–1148.

[33] T. Novak and A. Treytl, "Functional safety and system security in automation systems—A life cycle model," in *Proc. IEEE Int. Conf. Emerg. Technol. Factory Autom.*, 2008, pp. 311–318.

[34] T. Novak and A. Gerstinger, "Safety- and security-critical services in building automation and control systems," *IEEE Trans. Ind. Electron.*, vol. 57, no. 11, pp. 3614–3621, Nov. 2010.

[35] K. Sørby, "Relationship between security and safety in a security-safety critical system: Safety consequences of security threats," Ph.D. dissertation, Dept. Mech. Ind. Eng., Norwegian Univ. Sci. Technol., Trondheim, Norway, 2003.

[36] K. Stølen *et al.*, "Model-based risk assessment in a component-based software engineering process," in *Business Component-Based Software Engineering*. Boston, MA, USA: Springer, 2003, pp. 189–207.

[37] B. Hunter, "Integrating safety and security into the system lifecycle," in *Proc. Improving Syst. Softw. Eng. Conf.*, 2009, p. 147.

[38] R. Winther, O. A. Johnsen, and B. A. Gran, *Security Assessments of Safety Critical Systems Using HAZOPs* (ser. Lecture Notes in Computer Science), vol. 2187. Berlin, Germany: Springer, 2001, pp. 14–24.

[39] C. Schmittner, T. Gruber, P. Puschner, and E. Schoitsch, *Security application of Failure Mode and Effect Analysis (FMEA)* (ser. Lecture Notes in Computer Science), vol. 8666. Cham, Switzerland: Springer, 2014, pp. 310–325.

[40] *GSN Community Standard Version 1*, Origin Consulting, York, U.K., 2011.

[41] S. Lautieri, D. Cooper, and D. Jackson, "SafSec: Commonalities Between Safety and Security Assurance," in *Constituents of Modern System-Safety Thinking*. London, U.K.: Springer, 2007, pp. 65–75.

[42] N. Subramanian and J. Zalewski, "Quantitative assessment of safety and security of system architectures for cyberphysical systems using the NFR approach," *IEEE Syst. J.*, vol. 10, no. 2, pp. 397–409, Jun. 2016.

[43] I. Nai Fovino, M. Masera, and A. De Cian, "Integrating cyber attacks within fault trees," *Rel. Eng. Syst. Saf.*, vol. 94, pp. 1394–1402, 2009.

[44] M. Steiner and P. Liggesmeyer, "Combination of safety and security analysis—Finding security problems that threaten the safety of a system," in *Proc. 32nd Int. Conf. Comput. Saf., Rel. Secur.*, 2013, pp. 1–8.

[45] R. Kumar and M. Stoelinga, "Quantitative security and safety analysis with attack-fault trees," in *Proc. IEEE Int. Symp. High Assurance Syst. Eng.*, 2017, pp. 25–32.

[46] H. Abdo, M. Kaouk, J. M. Flaus, and F. Masse, "A safety/security risk analysis approach of industrial control systems: A cyber bowtie—Combining new version of attack tree with bowtie analysis," *Comput. Secur.*, vol. 72, pp. 175–195, 2018.

[47] M. Bouissou and J. L. Bon, "A new formalism that combines advantages of fault-trees and Markov models: Boolean logic driven Markov processes," *Rel. Eng. Syst. Saf.*, vol. 82, pp. 149–163, 2003.

[48] S. Kriaa, M. Bouissou, F. Colin, Y. Halgand, and L. Pietre-Cambacedes, *Safety and Security Interactions Modeling Using the BDMP formalism: Case Study of a Pipeline* (ser. Lecture Notes in Computer Science), vol. 8666. Cham, Switzerland: Springer, 2014, pp. 326–341.

[49] S. Kriaa, M. Bouissou, and L. Pièrre-Cambacédès, "Modeling the Stuxnet attack with BDMP: Towards more formal risk assessments," in *Proc. 7th Int. Conf. Risks Secur. Internet Syst.*, 2012, pp. 1–8.

[50] B. Kaiser, C. Gramlich, and M. Förster, "State/event fault trees-A safety analysis model for software-controlled systems," *Rel. Eng. Syst. Saf.*, vol. 92, pp. 1521–1537, 2007.

[51] M. Roth and P. Liggesmeyer, "Modeling and analysis of safety-critical cyber physical systems using state/event fault trees," in *Proc. 32nd Int. Conf. Comput. Saf., Rel. Secur.*, 2013.

[52] R. Mitchell and I. R. Chen, "Effect of intrusion detection and response on reliability of cyber physical systems," *IEEE Trans. Rel.*, vol. 62, no. 1, pp. 199–210, Mar. 2013.

[53] C. W. Ten, C. C. Liu, and G. Manimaran, "Vulnerability assessment of cybersecurity for SCADA systems," *IEEE Trans. Power Syst.*, vol. 23, no. 4, pp. 1836–1846, Nov. 2008.

[54] A. J. Kornecki, N. Subramanian, and J. Zalewski, "Studying interrelationships of safety and security for software assurance in cyber-physical systems: Approach based on Bayesian belief networks," in *Proc. Federated Conf. Comput. Sci. Inf. Syst.*, 2013, pp. 1393–1399.

[55] G. Sindre, "A look at misuse cases for safety concerns," in *Proc. Working Conf. Method Eng.*, 2007, pp. 252–266.

[56] G. Sindre and A. L. Opdahl, "Eliciting security requirements with misuse cases," *Requirements Eng.*, vol. 10, pp. 34–44, 2005.

[57] C. Raspotnig, P. Karpati, and V. Katta, *A Combined Process for Elicitation and Analysis of Safety and Security Requirements* (ser. Lecture Notes in Business Information Processing), vol. 113. Berlin, Germany: Springer, 2012, pp. 347–361.

[58] J. Jürjens, *Developing Saf.-Crit. Syst. With UML* (ser. Lecture Notes in Computer Science), vol. 2863. Berlin, Germany: Springer, 2003, pp. 360–372.

[59] J. Jürjens, *UMLsec: Extending UML for Secure Systems Development* (ser. Lecture Notes in Computer Science), vol. 2460. Berlin, Germany: Springer, 2002, pp. 412–425.

[60] J. Jürjens, "Developing safety-and security-critical systems with UML," in *Proc. DARP Workshop, Loughborough, U.K.*, 2003.

[61] G. Pedroza, L. Apvrille, and D. Knorreck, "AVATAR: A SysML environment for the formal verification of safety and security properties," in *Proc. 11th Annu. Int. Conf. New Technol. Distrib. Syst.*, 2011, pp. 1–10.

[62] J. Thomas, "Extending and automating STPA for requirements generation and analysis," Ph.D. dissertation, Eng. Syst. Division, Massachusetts Inst. Technol., Cambridge, MA, USA, 2013.

[63] I. Friedberg, K. McLaughlin, P. Smith, D. Laverty, and S. Sezer, "STPA-SafeSec: Safety and security analysis for cyber-physical systems," *J. Inf. Secur. Appl.*, vol. 34, pp. 183–196, 2017.

[64] A. Nourian and S. Madnick, "A systems theoretic approach to the security threats in cyber physical systems applied to Stuxnet," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 1, pp. 2–13, Jan./Feb. 2018.

Detecting Cyber-Attacks Against Cyber–Physical Manufacturing System: A Machining Process Invariant Approach

Zedong Li, Xin Chen, *Member, IEEE*, Yuqi Chen, Shijie Li, Hangyu Wang, Shichao Lv, and Limin Sun

***Abstract*—The era of the Industrial Internet of Things has led to an escalating menace of cyber–physical manufacturing systems (CPMSs) to cyber-attacks. Presently, the field of intrusion detection for CPMS has significant advancements. However, current methodologies require significant costs for collecting historical data to train detection models, which are tailored to specific machining scenarios. Evolving machining scenarios in the real world challenge the adaptability of these methods. In this article, We found that the machining code of the CPMS contains a complete machining process, which is an excellent detection basis. Therefore, we propose MPI-CNC, an intrusion detection approach based on Machining Process Invariant in the machining code. Specifically, MPI-CNC automates the analysis of the machining codes to extract machining process rules and key parameter rules, which serve as essential detection rules. Then, MPI-CNC actively acquires runtime status from the CPMS and matches the detection rules to identify cyber-attacks behavior. MPI-CNC was evaluated using two FANUC computer numerical control (CNC) machine tools across ten real machining scenarios. The experiment demonstrated the exceptional adaptability capability of MPI-CNC. Furthermore, MPI-CNC showed superior accuracy in detecting cyber-attacks against CPMS compared to existing state-of-the-art detection methods while ensuring normal machining operations.**

***Index Terms*—Computer numerical control (CNC), cyber attack, cyber–physical manufacturing systems (CPMSs), Industrial Internet of Things, intrusion detection.**

I. Introduction

MANUFACTURING industry is an important cornerstone of modern industrial development. With the advent of the Industrial Internet of Things and intelligent manufacturing, the global manufacturing industry is rapidly moving toward networked and intelligent development [1]. Computer numerical control (CNC) system is the core of cyber–physical manufacturing systems (CPMSs) that control the machining process of manufacturing equipment. CNC systems are widely used in important industries, such as the aviation industry, automobile manufacturing, and military industry. Tesla's Giga factory connects CNC systems to the industrial Internet to automatically control production processes, greatly improving production efficiency.

As an increasing number of factories integrate their CNC systems into the Industrial Internet, the security of CPMS has become a paramount requirement and faces formidable challenges. Intrusion detection approaches for CPMS have emerged as a prominent and burgeoning topic. Currently in the field of CPMS security, most researchers focus on training machine learning classification models to detect anomalies by analyzing side channel data, such as current [2], video [3], or audio [4], [5], [6], generated during machining. Some researchers have built digital twin models for CNC systems with a data-driven approach to do consistency checks on the runtime state of CNC systems to detect cyber-attacks [7]. There is also an offline approach to detect whether machining codes have been tampered with, which extracts digital features of machining codes and trains machine learning anomaly classification models [8]. These solutions can effectively detect anomalous processing behaviors for specific machining scenarios.

Regrettably, the absence of adequate security considerations for CNC system manufacturers has resulted in attackers being able to easily launch cyber-attacks by exploiting CNC system vulnerabilities, such as the lack of authentication mechanisms, plain-text transmission, and the existence of unfixed vulnerabilities in the system. Primarily, attackers target the machining code to introduce defects in the product processing. For instance, they may implant a Trojan into the firmware of the CNC system, surreptitiously tamper with the machining code passed into the system, and execute a malicious hole attack [9]. Additionally, attackers have demonstrated the use of steganography to tamper with machining code files in network traffic, diminishing the mechanical strength of the resulting product [10]. In a further form of attack, assailants manipulate key parameters in the memory of the CNC system. For instance, they substitute the processing material by tampering with parameters, integrating smart materials into gas masks to plant physical logic bombs. This causes gas masks to crack and leak during use [11]. Furthermore, attackers have utilized existing open-source tools like C3PO [12] and Industrial

Manuscript received 23 October 2023; revised 11 January 2024; accepted 22 January 2024. Date of publication 25 January 2024; date of current version 9 May 2024. This work was supported in part by the National Key Research and Development Program of China under Grant 110400ZG21. *(Corresponding author: Limin Sun.)*

Zedong Li, Xin Chen, Shijie Li, Hangyu Wang, Shichao Lv, and Limin Sun are with the Beijing Key Laboratory of IOT Information Security Technology, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100085, China, and also with the School of Cyber Security, University of Chinese Academy of Sciences, Beijing 101408, China (e-mail: lizedong@iie.ac.cn; chenxin1990@iie.ac.cn; lishijie@iie.ac.cn; wanghangyu@iie.ac.cn; lvshichao@iie.ac.cn; sunlimin@iie.ac.cn).

Yuqi Chen is with the School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China (e-mail: chenyq@shanghaitech.edu.cn).

Digital Object Identifier 10.1109/JIOT.2024.3358798

Security Exploitation Framework (ISF) [13] to send malicious instructions to CNC systems disrupting the processing processes. These instances underscore the pressing need for intrusion detection systems for CNC systems to mitigate such threats effectively.

Motivation: In practical production processes, the CNC system employs various machining codes to handle different products, leading to diverse machining scenarios. These different scenarios require distinct tool paths, raw materials, and machining tools, resulting in different side-channel features with audio, image, current, and voltage. To develop intrusion detection models using side-channel data across different machining scenarios, researchers typically need to gather side-channel data for each new scenario and repeat the training process, incurring significant time and labor costs. However, once attackers successfully deploy an attack script in a CPMS system, they can easily disrupt the different machining scenarios. Consequently, there is an urgent need for an adaptable intrusion detection approach within the CPMS system that can be readily deployed across a variety of machining scenarios to effectively counter existing attack methods.

Insight: Invariant rule-based detection is currently a popular method in the field of industrial control security to effectively detect anomalies due to cyber-attacks. Usually, industrial control devices execute control logic codes to control the normal operation of industrial systems based on the invariant control logic in the control logic codes. Researchers have utilized data-driven [14] or code-driven [15] approaches to extract control logical invariant rules in industrial control systems as the basis for intrusion detection in industrial control systems. They have achieved excellent detection results. Inspired by the invariant rule-based detection, we found that in the field of CPMS, the machining process of the CNC system is invariant, and the machining code contains comprehensive machining process invariant information. Therefore, the complete machining process invariant rules can be extracted by analyzing the machining code. The machining process invariant rules include key elements, such as machining trajectory and machining speed, which can comprehensively describe the machining process of the CNC system and is a reliable basis for intrusion detection.

Method: This article addresses the issue of the limited adaptive ability of the CPMS intrusion detection system by proposing MPI-CNC, an intrusion detection method based on Machining Process Invariant. MPI-CNC automatically and rapidly extracts detection rules from the machining code. The method first parses the machining code to extract tool paths, machining sequences, spindle speeds, and other key machining-related parameters as rules for detecting attacks. MPI-CNC then actively collects runtime machining status, and key parameters from the CNC system during machining. Finally, MPI-CNC verifies the consistency of the runtime machining data based on the detection rules to identify cyber-attacks.

Result: To verify the feasibility of the approach in this article, a prototype was developed based on the FANUC CNC system. We conducted experiments using real CNC machines, analyzed 10 real machining scenarios and 3 attack methods, and evaluated the deployment time cost, detection performance, and interference to the CNC system. Experiments demonstrated that MPI-CNC can be quickly applied to new machining scenarios without preprocessing and detect cyber-attacks accurately in runtime without affecting the normal operation of the CNC. MPI-CNC has better detection performance compared to the other state-of-the-art detection methods. The detection accuracy of machining code injection attack and parameter injection attack reaches 98.81% and 100%, respectively, while the best detection results of other methods are 98.38% [8] and 93.25% [7].

This article contributes as follows.

1) We propose a novel approach for the automatic extraction of detection rules by analyzing machining codes. It can rapidly generate detection rules for different machining scenarios, thereby improving the adaptability capability of CPMS intrusion detection.
2) We conducted a reverse analysis of the FOCAS protocol used in FANUC CNC systems and developed low-interference acquisition request packets that conform to the protocol format. This approach improves the efficiency of data acquisition while reducing interference to the machining process.
3) A prototype CPMS IDS was developed based on the FANUC CNC system. Although this prototype was developed for a specific CNC system, based on this idea, it can be modified to expand and adapt to other CNC systems and protocols.
4) We evaluated the adaptability capability and detection performance of our proposed approach in 10 real machining scenarios and 3 attack scenarios. Experiments show that this approach can be quickly applied to new machining scenarios without preprocessing and detect cyber-attacks accurately in runtime without affecting the normal operation of the CNC.

Roadmap: The remainder of this article is structured as follows. Section II briefly introduces the technical background related to CPMS, and Section III provides an overview of the MPI-CNC. Section IV details the MPI-CNC and the specific implementation. The experimental evaluation is detailed in Section V. Section VI introduces the related work of current CPMS intrusion detection methods. Section VII discusses the limitations of the MPI-CNC. Section VIII is the conclusion.

II. Background

This article is primarily dedicated to proposing an intrusion detection method for CPMS. This chapter serves to provide an overview of the research background, focusing on two essential aspects: 1) the composition and 2) machining process of CPMS, as well as the various forms of attacks encountered by CPMS.

A. Cyber–Physical Manufacturing Systems

The CPMS generally consists of an engineering station, a distributed numeric control (DNC) server, a machine data collection (MDC) server, and manufacturing equipment connected through an industrial switch [see Fig. 1(a)]. The

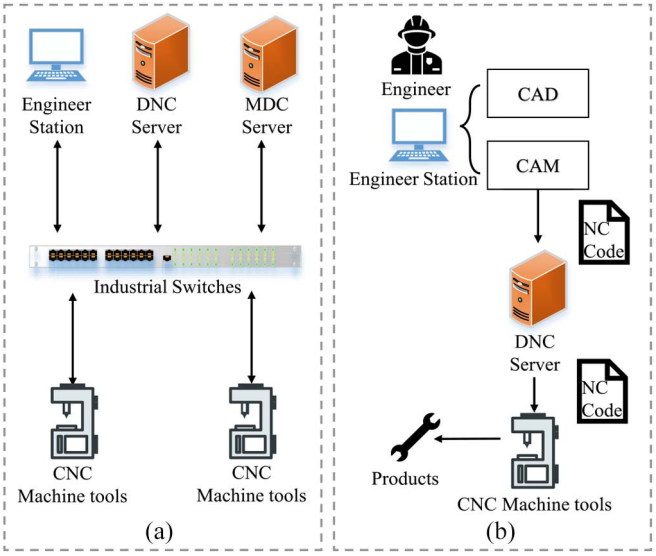

Fig. 1. Network topology and processing process of a CPMS. (a) Network topology of CPMS. (b) Processing process of CPMS.

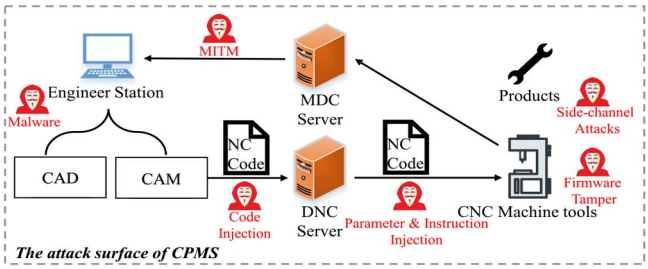

Fig. 2. Attack surface of CPMS.

engineering station is usually an office computer equipped with computer aided design (CAD), and computer aided machining (CAM) programs. The Processing process is shown in Fig. 1(b). Engineers utilize engineering design software to generate machining code (alternative name NC code), which is then uploaded to the DNC server. The DNC server distributes the NC code to the appropriate manufacturing equipment. The CNC system automatically controls the machining process by parsing the NC code. The MDC server interacts with the manufacturing equipment to collect various states of the equipment, including position, speed, temperature, and other information. This data is returned to the monitoring program of the engineering station, allowing engineers to monitor the machining process. Additionally, engineers can send control commands to perform runtime operations during the machining process.

B. Attack Model

In recent times, scholars have analyzed and categorized cyber-attacks targeting CPMS [16], [17]. This article investigates recent cyber-attacks against CPMS, analyzing the attack surface from the perspective of the production process (see Fig. 2). In the production process, the engineer station is connected to a local area network or even the Internet. Attackers can maliciously target the engineer station using spear-phishing [18], BadUSB [19], and other vectors carrying malicious code to exploit system and software vulnerabilities, stealing and tampering with CAD models and NC codes. Devices, such as engineer stations and DNC servers, typically communicate with the CNC system via Ethernet, using communication protocols that often lack authentication, encryption, and other security mechanisms. For example, DNC servers may use the FTP protocol to transmit NC code in clear text. Attackers can perform man-in-the-middle attacks [20], tampering with and stealing NC code from network traffic, and replaying network packets to inject malicious commands and parameters. The attacker can also tamper with the CNC system firmware [9], [21] to interfere with normal processing. However, such attacks are more difficult and require a deep understanding of the underlying code structure of the CNC system. As CPMS is a typical cyber–physical system [22], attackers can use side-channel attack methods, such as electrical measurement interference and acoustic resonance, to interfere with normal processing [23], or infer the production state of the machine tool and workpiece geometry information from leaked physical information [24], achieving a steganography attack.

The attacks were classified into three categories: 1) machining code injection; 2) parameter injection; and 3) instruction injection.

Machining Code Injection: Machining Code injection attack [9], [10], [25] refers to tampering with or replacing the machining code, the NC code, of the CNC system. By modifying key code segments, such as the machining path, spindle speed, or auxiliary control code, attackers can interfere with and disrupt the CNC machining process.

Parameter Injection: Parameter Injection [11], [26] refers to tampering with the parameters of the CNC system. There are many important parameters in the CNC system that affect the machining process, such as the spindle speed ratio value, the rapid feed rate value, and the alarm shielding. Therefore, if attackers can tamper with these key parameters, it will cause serious damage to the CNC system, affecting machining accuracy and potentially damaging the CNC machine.

Instruction Injection: Instruction Injection refers to sending malicious control commands to the CNC system, which disrupts the normal machining process. McCormack et al. [12] introduced an open-source tool called C3PO, which analyzes potential vulnerabilities in network services of 3-D printers and uses network vulnerabilities to send malicious commands to attack remote-controlled CNC systems. Attackers can also use the ISF [13] to inject malicious commands by sending attack scripts to disrupt the production process.

Moreover, the CNC systems also face security threats, such as physical cross-domain attacks and side-channel information leakage [27]. Nevertheless, it is pertinent to note that these threats lie outside the purview of this article. Their inclusion is excluded due to factors, such as their diminished feasibility, limited potential for harm, or their susceptibility, to detection by existing IDSs.

III. Overview

In this research, we propose an innovative intrusion detection method for CPMS, denoted as MPI-CNC. As illustrated in

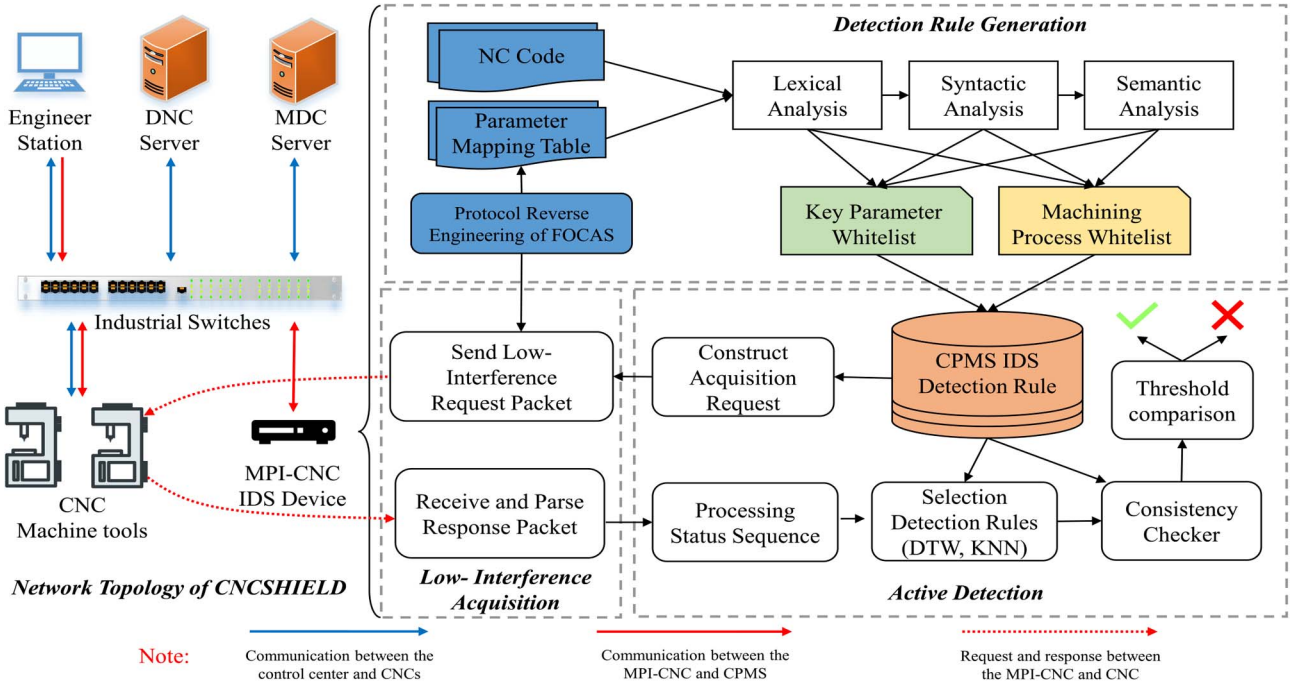

Fig. 3. Systematic approach to build MPI-CNC.

Fig. 3, this method comprises three distinct stages: 1) detection rule generation; 2) low-interference state acquisition; and 3) active detection. In this section, we present a concise overview of the fundamental framework for analyzing intrusion detection methods.

Detection Rule Generation: The detection rule generation module employs static analysis to parse the NC code and extract detection rules. The NC code encapsulates complete machining processes, including key parameter rules and machining process rules. The key parameter rules are utilized to monitor and verify vital parameters within the CNC system, ensuring their accuracy, stability, and safeguarding against malevolent tampering that could lead to diminished machining precision or machine malfunctions. On the other hand, the machining process rules establish reference guidelines by analyzing the invariant characteristics of machining processes in the NC code, enabling the detection of malicious attacks, such as tampering with machining trajectories or program substitution. The extracted detection rules from the NC code furnish a comprehensive depiction of the machining process and enable proactive detection of unexpected anomalies and network attacks.

Low-Interference Acquisition: The low-interference state acquisition module is responsible for the runtime collection of machining states within the CNC system. Typically, CNC system manufacturers provide monitoring software or development kits for monitoring the system's operational status. For instance, the FANUC Focas 1/2 development component facilitates secondary development. It enables runtime remote monitoring through active communication with the CNC system. The development component provides essential information, such as NC programs, tool positions, and spindle speeds. However, direct use of the original development kit for high-frequency data collection can increase the network load of the CNC system, negatively impacting normal machining operations and real-time performance. To circumvent this issue, our study employs reverse engineering to analyze proprietary protocols. We also customize data collection requests and eliminate redundant ones. As a result, we achieve low-interference high-frequency acquisition of runtime machining states within the CNC system.

Active Detection: The active detection module primarily identifies anomalies in the machining process. It processes the runtime machining state data collected by the low-interference state acquisition module and generates alerts. It verifies whether the machining state of the CNC system adheres to the key parameter rules and the machining process rules. This approach prevents code tampering, manipulation of key parameters, and malicious instruction attacks. The active detection stage necessitates determining two critical monitoring parameters: 1) error threshold and 2) monitoring window size. Initially, we experiment with multiple monitoring window sizes based on the state acquisition frequency to determine the optimal size. Subsequently, under specific window sizes, we calculate the cumulative normal error for each monitoring window and set the error threshold using the maximum observed error.

IV. Approach

A. Problem Statement

The CNC manufacturing process is a complex industrial control process in which the CNC system performs closed-loop control of the relative motion of the tool and the workpiece based on multiple sensor data. In this article, we use $u(t)$ in (1) to describe the machining state of the CNC at time

t, where $P(x, y, z)$ indicates the coordinates of the tool in the xyz three axes, $S(t)$ indicates the spindle speed, $F(t)$ indicates the feed rate, and $T(t)$ indicates the current tool number

$$u(t) = (P(x, y, z), S(t), F(t), T(t)). \tag{1}$$

We define (2) with $r(n)$ to describe the machining process indicated by the machining code, which represents the invariant characteristics of the CNC machining process. Specifically, we use $F(x, y, z)$ to represent the curve equation of the machining path, which is commonly straight lines and circular arcs. Start(x, y, z) and End(x, y, z) represent the start and end points of the machining path. The combination of machining path, spindle speed, feed rate, and tool number provides a complete description of the machining process

$$r(n) = (F(x, y, z), \text{Start}(x, y, z), \text{End}(x, y, z), S(n), F(n), T(n)). \tag{2}$$

We use $\mathcal{M}()$ in (3) to describe the CNC machining model, where $\varepsilon(t)$ represents the internal losses of the CNC and reasonable errors due to natural factors

$$u(t + 1) = \mathcal{M}(u(t), r(t), \varepsilon(t)). \tag{3}$$

In the normal machining process, the CNC machining state has reasonable errors $\varepsilon(t)$ due to machine wear and tear, current and voltage jitter, and other factors. However, when the CNC is under a cyber-attack, it can deviate significantly from the CNC machining model $\mathcal{M}()$ and violate the current machining process $r(n)$. Therefore, in this article, we designed an intrusion detection method based on the invariant characteristics of the CNC machining process. Our approach is divided into a detection rule generation module, a low-interference acquisition module, and an active detection engine [see Fig. 3].

B. Intrusion Detection Rule

The machining process is a crucial basis for manufacturing and processing workpieces. The NC code contains the most complete and comprehensive machining process information, such as spindle speed, feed rate, and machining path. The CNC system interprets the NC code into executable instructions to control the various components of the CNC machine tool to complete the machining operations. Through the analysis of the NC code, the following intrusion detection rules can be generated: key parameter whitelist rules and machining process whitelist rules. The approach of generating detection rules based on code analysis demonstrates great applicability in the industrial control field [15], [28].

1) Key Parameter Whitelist Rule: In CNC systems, there are numerous important parameters that can affect the actual production process. The process of sending commands from the CNC system to control the hardware needs to be adjusted to the specific parameters. For instance, the CNC system adjusts the tool's landing position and movement trajectory during the actual machining process based on parameters, such as tool radius compensation and length compensation, or the CNC system controls the feed acceleration based on parameters related to acceleration and deceleration. These parameters directly impact the machining accuracy and stability of the machine tool. If the key parameters in the CNC system are maliciously tampered with by attackers, it can result in decreased machining accuracy or even machine tool failure. Typically, key parameters in CNC systems have specific values or value ranges. For instance, specific tool radius compensation and length compensation parameters have fixed values, and the control parameters for rapid feed acceleration and deceleration generally fall within the range of 140–160 ms. Therefore, we analyze the parameters of FANUC CNC in Table I, and establish whitelist rules for key parameters and their value ranges to monitor the correctness of the key parameters in the CNC system.

TABLE I
FANUC PARAMETERS MAPPING TABLE

Parameter Type	FOCAS Address Mapping	Data Type	Number of parameters
Tool Compensation Parameters	0x000800000001 -0x000800000190	Real	400
Macro Variables	0x001500000001 -0x0015000003e7	Real	633
CNC Parameter	0x008d00000001 -0x008d00006bd9	Bit(axis), Byte(axis), Word(axis), Real(axis)	27609
PMC Parameter	0x800100000000 00000000 -0x800100000bb7 00000009	Bit(axis), Byte(axis), Word(axis), Real(axis)	6572
All Parameters			35214

2) Machining Process Whitelist Rule: The International Organization for Standardization (ISO) has established ISO-6983-1 [29] as the international standard for CNC programming languages. This standard delineates the lexical and syntactic rules governing CNC codes, thereby forming a programming language comprising G-codes and M-codes. Numerous CNC system manufacturers have introduced CNC control products adhering to the ISO-6983-1 standard. For instance, SIEMENS CNC systems, such as SINUMERIK 802D and SINUMERIK 840D, as well as FANUC CNC systems like 0i-md and 0i-mf, support NC programming in compliance with this standard. While CNCs from various manufacturers or models may exhibit diverse representations of NC code programming, they share commonalities, and the discrepancies in syntax and programming concepts are essentially minimal. Consequently, leveraging our proposed detection scheme and considering these shared characteristics, we can customize the intrusion detection system to be applicable to different models of CNC systems.

The CNC system controls the machining process through the NC code. The spindle and multiple servo axes in the CNC system work in coordination to control the movement and rotation of the tool and workpiece, completing automated production machining. Therefore, we parse the NC code and extract the invariant relationships of tool motion trajectories, feed rates, spindle speeds, and other parameters for each step of the machining process flow to generate machining process rules. These rules serve as benchmarks for detecting malicious attacks, such as NC code tampering or parameter injection.

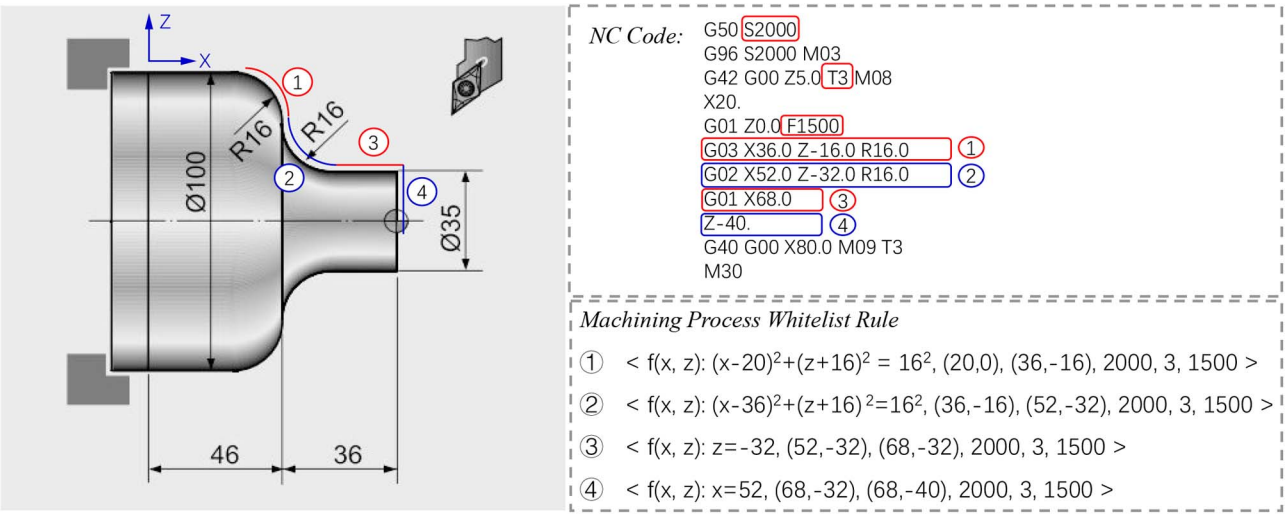

Fig. 4. Case of machining process whitelist rule.

Taking turning machining as an example [see Fig. 4], we demonstrate how to generate detection rules based on the invariance of the machining process using NC codes. The CNC system executes the NC codes to automatically control the machining process during turning. The NC codes specify the spindle speed, feed rate, and tool number for the machining process, and then use G codes to specify the tool's movement trajectory, such as common linear machining($G01$) and circular machining($G02, G03$). Therefore, we perform lexical, syntactic, and semantic analysis on the NC codes to generate the machining process rules, such as rule ①, which indicates that under the condition of spindle speed $S = 2000$ and feed rate $F = 1500$, tool number 3 moves along the curve $(x-20)^2 + (z+16)^2 = 16^2$, with a starting point of $(20, 0)$ and an ending point of $(36, -16)$.

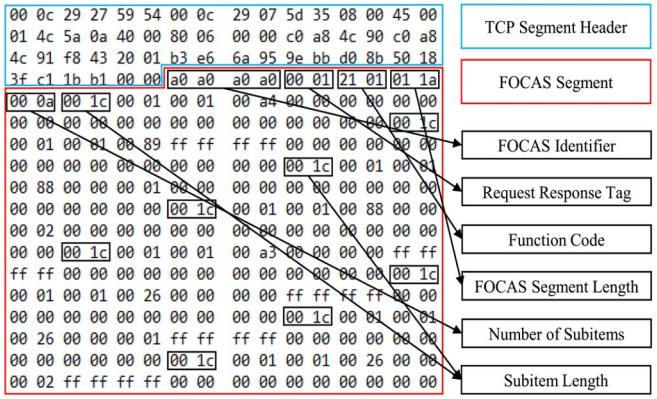

Fig. 5. FOCAS protocol reverse analysis results. This figure shows a binary request packet for collecting the current machining coordinates of a CNC using Focas, which consists of a Focas protocol header and ten subitems.

C. Low-Interference Acquisition

In order to collect the runtime status of the CNC, the conventional method is to use the communication interface provided by the CNC manufacturer. However, we found that the acquisition frequency of Focas, the communication interface provided by FANUC, is too low, which leads to an increase in the alarm delay for intrusion detection and affects the accuracy of the detection rule selection. For this reason, we manually reverse analyzed the protocol format of Focas and designed low-interference acquisition packets.

1) FOCAS Protocol Reverse Engineering: We capture mirror traffic on an industrial switch and conduct reverse protocol analysis on the proprietary protocol Focas for FANUC CNC systems. Focas protocol is an application-layer protocol based on TCP/IP. During the process of establishing a connection, the Focas protocol requires two rounds of TCP handshake to establish the connection. First, the client uses port A (any available port) to initiate a connection establishment request to port 8139 of the CNC system. Then, the client establishes a second connection to port 8193 of the CNC system using port A + 1 or A + 2. Subsequent request and response operations are performed on port A + 1 or A + 2. Reverse protocol analysis reveals the frame format of Focas protocol [see Fig. 5]. The first 4 bytes of the payload section are always a0a0a0a0, serving as the identification for Focas protocol. The 5th and 6th bytes represent the request/response flag. The Focas function code is located in bytes 7 and 8. The 9th and 10th bytes represent the length of the payload data. The 11th and 12th bytes indicate the number of subitems. The first 12 bytes form the header of the Focas protocol. The subitems in the Focas protocol include subitem length, fixed padding, and subfunction code. The payload information of the subitem includes the request parameter address and data format, which are not explicitly described in this article to prevent potential misuse by malicious individuals.

2) Constructing Low-Interference Acquisition Packages: Based on the results of the reverse protocol analysis mentioned above, we found that when using the API interface functions provided by Focas to collect position coordinates, spindle speed, and feed speed of CNC systems, multiple Focas request packets need to be sent to collect the machining status of the CNC system at the same moment. Moreover, these request packets usually contain irrelevant subitems that are unrelated

to attack detection. Under the high-frequency collection, these irrelevant subitems consume a significant amount of network and CNC system computing resources, which affects the CNC system and reduces detection efficiency. To solve this problem, we extracted the detection-related subitems from multiple request packets and combined them into a single request packet, which is then sent to the CNC system to collect multiple machining statuses at the same moment. Upon receiving the response packet, the status data of the CNC system is extracted based on the Focas protocol frame format.

The low-interference acquisition packages based on reverse engineering of the proprietary protocol greatly reduce the network overhead of status collection and minimize the interference on the CNC system. Furthermore, the analysis shows that the S7comm-nck protocol used by SINUMERIK 828, and 848 CNCs can also be used to construct low-interference request packets using the methods in this article. Detailed experimental data can be found in Section V-D.

D. Active Intrusion Detection Method

In this section, we outline the specific methods used for detecting attacks on manufacturing processes based on detection rules [see Fig. 3]. Our approach employs a low-interference, runtime active intrusion detection technique that does not disrupt the normal CNC machining process. During the implementation of this module, we have effectively addressed two key challenges.

1) Common phenomena, such as circuit instability, mechanical jitter, and equipment aging, can occur during the machining process. These issues can lead to inconsistencies in the CNC's execution time for each instruction. As a result, it becomes challenging to accurately and promptly match the collected runtime machining state to the detection rules.
2) The introduction of jitter and other interference due to regular errors, which can lead to an increased false alarm rate in the detection program, necessitating the need to distinguish between regular errors and cyber attacks.

1) Selection Detection Rules: To address challenge 1, We employed the dynamic time warping (DTW) algorithm and the k-nearest neighbors (KNNs) algorithm.

DTW is a dynamic programming algorithm that measures the similarity between time series [30], particularly those of varying lengths. It is commonly used in the fields of speech recognition, gesture recognition, and information retrieval due to its applicability to temporal data. In our study, we utilized the DTW algorithm to align a reference state sequence with a rule label to a captured runtime processing state sequence, with timestamps arranged in chronological order.

KNN is a nonparametric method used in supervised learning [31]. KNN is based on a simple and intuitive concept: if the majority of the k-most similar samples in the feature space of a given sample belong to a certain category, then the sample is also classified as belonging to that category. The algorithm makes its decision by considering only the category of the nearest one or more samples. In our study, we employed the KNN algorithm to classify runtime processing state points. This allowed us to select the appropriate detection rules based on reference state sequences that were labeled with the rules.

2) Consistency Checker-Based Detection Windows and Thresholds: To address challenge 2, we implemented a detection window and alarm threshold in our approach. During the detection process, we collect the runtime state of continuous machining from the CNC machine tool for the duration of the window time and then accumulate the error between each runtime machining state and the machining process rules. An alarm is triggered when the accumulated error exceeds the threshold. If the window expires and the cumulative error does not exceed the threshold, the cumulative error is reset to 0 and a new inspection window is initiated. In this article, we employed (4) to conduct a consistency check, which involves accumulating the Euclidean distance between the runtime machining state and the machining process rule within the inspection window and comparing it with the inspection threshold. Specifically, as in (5), the actual error value is obtained by calculating the distance D between the actual position and the machining trajectory of the machining process rule, and the deviation of the actual feed rate F and spindle speed S from the machining process rule. Additionally, we performed dissimilarity verification between the runtime key parameter matrix $\mathbb{C}_{3\times n}$ and the key parameter rule $\mathbb{K}_{3\times n}$ to ensure key parameter consistency. Equation (3) serves as the theoretical foundation for our machining process consistency verification, enabling us to identify attacks, such as machining code injection, parameter injection, and instruction injection, on the CNC system during the machining process

$$\left\{\sum^{W\ \text{size}} \|y(t)-r(t)\| \le \delta(t)\right\} \wedge \{\mathbb{C}_{3\times n} \oplus \mathbb{K}_{3\times n} = [0]_{3\times n}\} \tag{4}$$

$$\|y(t)-r(t)\| = \sqrt{D^2 + (F-F_r)^2 + (S-S_r)^2}. \tag{5}$$

The active detection engine detects whether the CNC is under attack in an active and low-interference way, and its core part is shown in Algorithm 1. It first establishes a communication connection with the CNC (line 1); then parses the detection rules to simulate the machining path and constructs the active acquisition packet (lines 2–4) and then the attack is detected (lines 5–16); and, finally, when the detection is complete, the connection is disconnected (line 17). In the attack detection phase, the first step is to initialize the detection window and detection threshold (line 6). Next, a low-interference request packet is sent to the CNC, followed by receiving the response data and parsing the protocol to extract the processing state values (lines 7–10). Then, the DTW algorithm is used to match the detection rules for the data in the current detection window. Finally, a consistency check is done on the machining state and key parameters to see if the detection rules are satisfied (lines 12–14). Line 15 indicates the setting of the required frequency for the CNC to achieve low interference.

V. Evaluation

In this section, we focus on answering the following research question.

Algorithm 1: Algorithm of Active Detection

Input : Key Parameter Rule
Machining Process Rule
Detection Window Size
Output: Cyber-Attack Alerts

```
Connect(cncIP, cncPORT);
RuleDB ← LoadRules(KeyParameter, MachiningProcess);
PathSim ← PathSimulation(RuleDB);
AcquisitionPKG ← PackageConstructor(RuleDB);
while ProcessFlag do
    InitDetectionWindow();
    for 0 to DetectionWindowSize do
        Send(AcquisitionPKG);
        ProcessingStatus ← Receive();
    end
    FlaggedStatus ← DTW(ProcessingStatus, PathSim);
    if ConsistencyChecker(FlaggedStatus, RuleDB) is
    False then
        Alarm("Illegal Processes: cncIP, RuleNo.");
    end
    Sleep(t);
end
Disconnect();
```

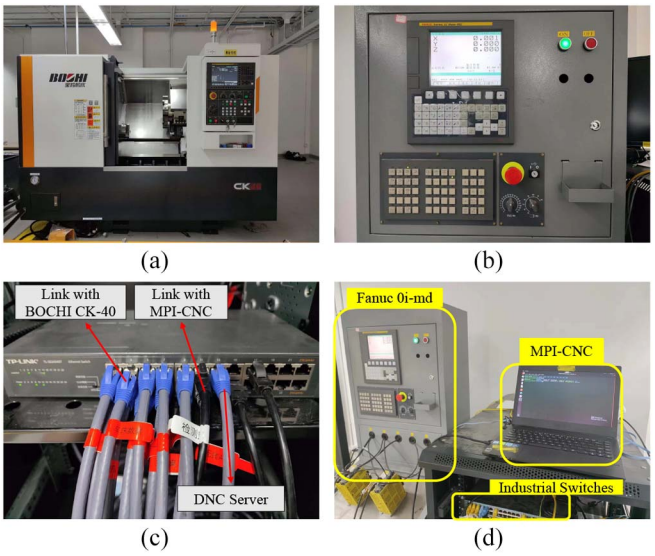

Fig. 6. MPI-CNC experimental environment. (a) BOCHI CK-40 with Fanuc 0i-tf. (b) Fanuc 0i-md. (c) Industrial switches. (d) MPI-CNC deployment environment.

RQ1: What is the time cost of MPI-CNC to generate rules?

RQ2: What is the effectiveness of MPI-CNC detection against cyber attacks on CNC systems?

RQ3: Does MPI-CNC affect CNC machine tool machining efficiency?

We used 3589 lines of C code and 2492 lines of Python code to implement MPI-CNC. MPI-CNC is deployed on a ThinkPad P15Gen2 with an 8 cores Intel Core i9-11950H CPU and 64 GB of RAM. MPI-CNC was evaluated using Fanuc 0i-md CNC and Fanuc 0i-tf CNC.

A. Experimental Environment

Experimental Design: To answer the three research questions above, experiments were conducted using real machining environments with the FANUC CNC system. Due to the lack of real-world cyber-attack data for the FOCAS CNC system, three attack methods discussed in Section II-B were implemented, and the effects of the attacks were demonstrated on real equipment. First, multiple NC programs from realistic machining scenarios that are applicable to the FANUC CNC system were analyzed, and detection rules were generated to verify the accuracy of the automatic parsing of NC code for rule generation. Second, the proposed intrusion detection method was compared with other detection models to evaluate its performance in detecting network attacks in the NC machining process. Finally, in order to demonstrate that our solution has minimal impact on the CNC system's machining process, the variations in machining time were monitored during the detection phase, and the network resource utilization was compared between using low-interference data collection requests and using FOCAS standard interface for collecting CNC system's machining status.

Experimental Environment: In this work, we conducted experiments using a FANUC 0i-tf CNC system [Fig. 6(a)] and FANUC 0i-md CNC system [Fig. 6(b)]. as shown in Fig. 6, We connect the MPI-CNC to the industrial switch connected to the CNC and configure its IP address to be in the same network segment as the other devices so that it can communicate with the CNC normally. Fig. 6(c) partly shows the connection status of BOCHI CK-40, DNC server, and MPI-CNC to each port of the industrial switch. Fig. 6(d) shows the site layout of the Fanuc 0i-md intrusion detection experimental environment, including the CNC, MPI-CNC, and industrial switch.

Cyber-Attack Setting: Specifically, we discuss three attack methods against CNCs in this work, which were implemented on FANUC CNCs due to the lack of available attacks for evaluation purposes. The first attack method is the machining code injection attack, which involves injecting malicious machining instructions and machining paths into the NC code. We introduced 20 tamperings in 10 different machining codes, including creep attacks and trajectory scaling, to evaluate the detection capability of the methods. The second attack method is the remote parameter injection attack, which was implemented by tampering with key CNC parameters through request packets sent to the FANUC CNC based on the Focas protocol inversion results. The third attack method is the malicious command injection attack, which involves tampering with the designated ports of the PMC by sending request packets to the FANUC CNC based on the Focas protocol inversion results and the CNC's interface manual. This allows for malicious commands, such as remote start/stop and on/off coolant, to be injected.

B. RQ1—Time Cost and Accuracy of Generating Detection Rules

We collected 57 NC codes applicable to FANUC CNC systems from real machining scenarios and Internet platforms, such as Github and Traceparts. These codes involve turning and milling processes and consist of 8354 instructions, including instructions for linear machining, circular arc machining,

TABLE II
TIME COST OF GENERATING DETECTION RULES

NC Code	Code Lines	Number of Rules	Times Cost(ms)
O5665-NC	134	90	1.004
O6383	150	93	0.805
NCViewer.nc	5780	5753	0.962
NCtest26.NC	64	61	0.768
7190.3-1A.nc	255	222	0.792
...	...	...	...
Number of NC Code: 57	Total Code Lines: 8354	Total Rules: 7671	Totle Time Cost: 53.579ms

TABLE III
COMPARING THE ACCURACY AND TIME COST OF DTW AND KNN ALGORITHMS IN SELECTING DETECTION RULES

Algorithms	KNN				DTW
	k=3	k=5	k=7	kd-tree	
Accuracy %	96.95	97.82	96.83	97.32	99.38
Times Cost(ms)	273	321	326	137	352

tool changing operations, coolant control, and more. By analyzing the machining instructions in these NC programs, we extracted attack detection rules. Unlike other high-level programming languages, NC programs are not as complex, typically consisting of multiple G codes and M codes. We used the number of G codes that control the machining trajectory to represent the size of the program and generated several key parameter rules based on the relationship between G codes and M codes, as well as one machining process rule per G code.

First, we analyzed the FANUC CNC system user manual and interface manual and combined the results of reverse engineering the FOCAS proprietary protocol to establish a mapping table of G codes, M codes, and system parameter addresses. Then, based on the semantic information of G and M codes in the NC codes, we selected important parameters related to machining and generated key parameter rules. Next, we parsed the G codes that control the machining trajectory, abstracted the curve equation of the machining trajectory based on its semantic information, and combined information, such as the starting point and ending point of the machining, spindle speed, feed rate, and tool number to generate machining process rules.

As shown in Table II, we analyzed all the collected NC codes and recorded all the generated key parameter rules and machining process rules. The 57 NC codes we collected totaled 8354 lines of machining instructions. For these collected 57 NC codes, key parameters related to machining are identified and a total of 6056 machining process rules are generated. In the process of generating inspection rules, we recorded the number of lines of machining code, the number of inspection rules generated, and the time cost of generating the rules for each NC code. The time taken to analyze the generation of inspection rules for a single NC code is 0.94 ms on average, and the time taken for a single inspection rule is 0.007 ms [Table II]. To verify the correctness of the generated rules, we selected 10 representative NC codes and manually verified the accuracy of the automatically generated detection rules using all the key parameter rules and machining process rules. The results showed that the accuracy of the detection rules in a limited number of NC program samples was 100%.

Answer RQ1: The proposed method in this article automatically analyzes NC code and generates comprehensive detection rules without relying on historical manufacturing data. Each NC code takes 0.94 ms to generate accurate inspection rules. The time cost of the method in this article is extremely low and can be quickly used in new machining scenarios.

C. RQ2—CPMS Cyber-Attack Detection Results

In Section II-B, we provide a comprehensive review of the current state-of-the-art research on attacks against manufacturing processes and a summary of three common attack methods. It is worth noting that publicly available CPMS attack methods or attack data sets are typically tailored to specific machining scenarios, devices, and processing processes, and there are currently no generic CPMS attacks. Therefore, in this work, we evaluate the detection effectiveness of the MPI-CNC by implementing three attack methods.

1) Selection Detection Rules Results: We conducted experiments to evaluate the accuracy and time cost of KNN and DTW algorithms in rule selection using 25 283 data points from real machining scenarios. The results are shown in Table III. We tested the performance of the KNN algorithm by setting different values of k in the rule selection experiments and using a kd-tree data structure. The test results showed that when k was set to 5, the rule selection accuracy was 97.82%, which was the optimal parameter for the KNN algorithm in rule selection. It is worth noting that the use of a kd-tree data structure greatly reduced the time cost, with rule selection for 25 283 data points taking only 137 ms. This makes it suitable for complex machining scenarios and real-time detection requirements. When using the DTW algorithm for rule selection, the selection accuracy was as high as 99.38% and the time cost was 352 ms, which falls within an acceptable range and meets real-time alarm requirements. In summary, to improve detection accuracy, MPI-CNC adopts the DTW algorithm as its rule selection algorithm. For complex machining scenarios with high-real-time detection requirements, the KNN algorithm based on a kd-tree data structure should be used.

2) Cyber-Attack Detection Results: We conducted experiments to evaluate the performance of different detection windows in attack detection, and the results are presented in Table IV. The active detection engine successfully detected the machining code injection attack, the key parameter injection attack, and the malicious instruction injection attack. These attacks interfere with the normal machining process and result in changes to the machining trajectory and machining state, which can be directly reflected in the machining process and the key parameters of the CNC system. The active detection engine actively communicates with the FANUC CNC to map its actual machining status and key parameters. This active detection approach makes it difficult for the attacks to be

TABLE IV
COMPARISON OF CYBER-ATTACK DETECTION RESULTS IN DIFFERENT DETECTION WINDOWS AND DETECTION THRESHOLD CASES

Window size /Threshold	Detection accuracy	False alarm rate	Missing alarm rate	Alarm delay(s)
10/0.1	99.15%	2.89%	1.83%	1.45
50/0.5	**98.81%**	**1.09%**	**2.42%**	**2.45**
100/1	98.68%	0.75%	6.17%	3.71
200/2	96.88%	2.61%	7.15%	6.32
500/5	94.87%	4.44%	6.25%	13.75

TABLE V
COMPARISON WITH OTHER CPMS IDS

Attack Type	Machining Code Injection	Parameter Injection	Instruction Injection
Digital Twins[7]	N/A	93.25%	93.25%
KCAD[4]	81.39%	81.39%	N/A
LTDT[3]	95.55%	95.55%	N/A
LSTM-AE[6]	94.79%	94.79%	N/A
Machining Code Analysis[8]	98.38%	N/A	N/A
MPI-CNC	**98.81%**	**100.00%**	**100.00%**

hidden, as it requires the attacker to gain insight into the real machining process and manipulate the CNC firmware, tamper with the network communication module, or employ other sophisticated methods to feedback network data that conforms to the machining process rules and key parameter rules.

As shown in Table IV, we set different detection window sizes of 10, 50, 100, 200, and 500 in the active detection experiments. We set the alarm threshold in millimeters based on the machining accuracy of the CNC machine tool of 0.01 mm multiplied by the window size. The test results show that the detection accuracy of the method proposed in this article is 99.15% with a detection window of 10, and the accuracy decreases with the increase of the detection window, down to 94.87%. The reason for this phenomenon is that as the detection window increases, the detection threshold increases, increasing the missing alarm rate and a decrease in detection accuracy. In Table IV, the missing alarm rate is 1.83% when the detection window is 10. The larger the detection window, the larger the missing alarm rate, and when the detection window is 500, the missing alarm rate is 6.25%. It is worth noting that the false alarm rate becomes larger when the detection window is too large and too small. The alarm delay increases as the detection window increases, mainly because the DTW algorithm takes more time to match more data. Considering the above, we conclude that the optimal detection window size of this method is 50, the detection accuracy is 98.81%, the false alarm rate is 1.09%, the missing alarm rate is 2.42%, and the alarm delay is 2.45 s.

3) Comparison With Other CPMS IDS: In order to demonstrate the effectiveness of our detection scheme, MPI-CNC was compared to its performance with other CPMS ICS models. Specifically, we compared with representative models that are commonly used for detecting manufacturing process attacks, namely, digital twin-based intrusion detection [7], side-channel analysis-based intrusion detection (KCAD [4], LTDT [3], LSTM-AE [6]), and machining code analysis-based intrusion detection [8]. The digital twin-based intrusion detection models require historical processing state data for fitting digital twin models. KCAD and LSTM-AE collect audio data generated by manufacturing equipment during processing to learn anomaly classification models. LTDT analysis of processing video classification anomalies. The machining code analysis-based intrusion detection extracts features of NC codes to train SVM models for offline classification of anomaly codes.

We launched 200 machining code injection attacks against 10 different machining scenarios (machining codes), each introducing 20 tampering points (such as replacement of G02 and G03 with G01; insertion of protrusions or depressions; and modification of endpoints to cause deformation).In addition, 100 parameter injection attacks and command injection attacks were used to test the detection performance of the MPI-CNC. Detection results and attack logs are collected and used to calculate detection accuracy, miss rate and false alarm rate of MPI-CNC, as in

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}} \tag{6}$$

$$\text{Missing Alarm} = \frac{\text{FN}}{\text{TP}+\text{FN}} \tag{7}$$

$$\text{False Alarm} = \frac{\text{FP}}{\text{TN}+\text{FP}}. \tag{8}$$

The methods have achieved high-detection accuracy using current state-of-the-art techniques, as shown in Table V. The detection accuracy of the digital twin-based detection method against parameter injection attacks and instruction injection attacks is 93.25% [7]. The detection accuracy of the KCAD against processing code tampering and instruction injection attacks is 81.39% [4]. In addition, the LTDT and LSTM-AE models have high-detection accuracies of 95.55% [3] and 94.79% [6]. However, the side channel data features behave differently in different processing scenarios, which makes it difficult to apply to new scenarios quickly. The code analysis-based intrusion detection directly analyzes processing code for offline detection with 98.38% detection accuracy [8], but it lacks runtime detection capability during processing. Compared with the above methods, MPI-CNC can cope with a wider range of attack scenarios. Since we analyze the key parameters of the CNC system and actively detect the key parameter information during the machining process, we can detect parameter injection attacks and instruction injection attacks by 100%. Also, the method in this article analyzes the NC code to generate comprehensive detection rules, so it has a higher detection accuracy similar to the machining code analysis method, with a detection accuracy of 98.81%.

Answer RQ2: Experiments have proved that MPI-CNC can cope with three attack methods, which is more comprehensive than the traditional detection model based on historical data. Moreover, MPI-CNC is significantly better than other detection methods in terms of accuracy of 98.81%.

D. RQ3—Low-Interference Experimental Results

The computational performance of CNC systems is not as high as that of traditional PCs. Therefore, we need to be cautious about whether active detection methods will affect the normal operation of NC systems. In this article, we adopt a low-interference, polling-based approach to collect the

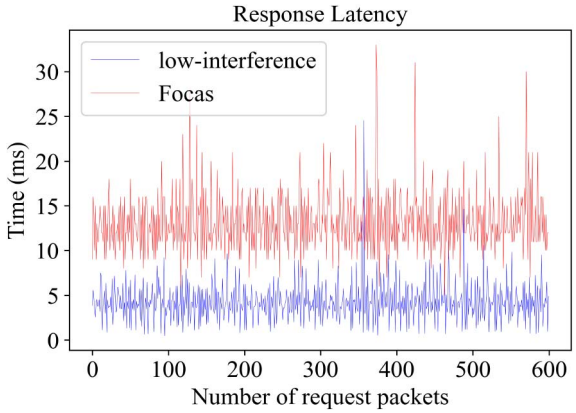

Fig. 7. Comparison of response latency for low interference and Focase.

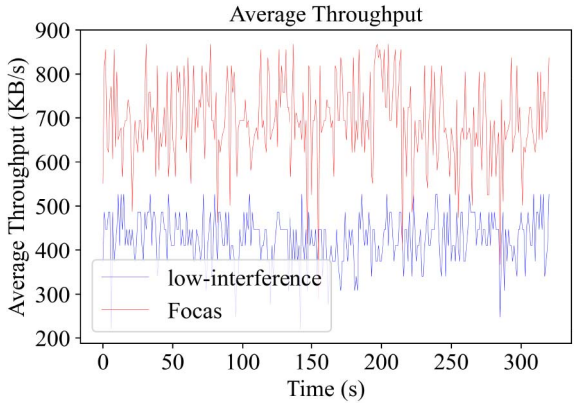

Fig. 8. Comparison of average throughput for low interference and Focase.

machining status of NC systems, and then process the data and detect attack behaviors remotely. In this section, we compare the acquisition delay, network resource utilization, and impact on machine tool processing times between the low-interference collection method proposed in this article and the FOCAS standard interface for collecting the machining status of CNC systems.

1) Acquisition Latency and Network Throughput: As shown in Fig. 7, the average response time for a single collection using the conventional FOCAS collection method is 13.240 ms, with a collection frequency of 75.53 times per second. Using the low-interference collection method proposed in this article, the average response time for a single collection is 4.368 ms, with a collection frequency of 228.94 times per second. The response delay is reduced by 67.00%, and the sampling frequency is increased by 203.12%. Meanwhile, in Fig. 8 the average throughput of the CNC system using the conventional FOCAS collection method is 692.04 KB/s, while the average throughput of the NC system using the low-interference collection method proposed in this article is 426.23 KB/s, resulting in a decrease in network throughput of 38.41%. Experiments have shown that using the self-assembled packet method based on private protocol reverse engineering proposed in this article for collecting the machining status of CNC systems reduces interference to the CNC systems while improving the collection frequency.

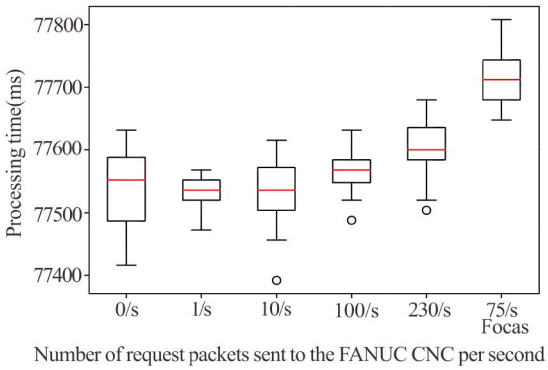

Fig. 9. Influence of network traffic on FANUC CNC.

2) Processing Times Under CPMS IDS: In order to demonstrate that our detection scheme has minimal impact on the machining time of CNC systems, we conducted experiments with different request frequencies during the active detection process to observe changes in machining time. To avoid the forwarding delay of industrial switches, we directly connected the CNC system with Ethernet cables and sent 1, 10, 100, and 230 low-interference request packets per second, as well as 75 Focas standard interface request packets per second. As shown in Fig. 9, under normal conditions without any external interference, the machining time of the CNC system was approximately 1 min and 17.545 s. When the CNC system was subjected to varying degrees of external interference, we found that its machining time was minimally affected. In the case of sending 230 packets per second, the machining time of the CNC system increased by only 0.076% and no packet loss was observed. This indicates that the FANUC CNC system has the capability to process at least 230 packets per second without affecting its normal operation. However, when using Focas standard interface request packets with a maximum rate of 75 requests per second, the machining time of the CNC system increased by 0.217%, which was significantly higher than the low-interference data collection method used in this article. Moreover, the FANUC system can handle a maximum of 75 requests per second. The experiment results show that our detection scheme has minimal impact on the machining time of CNC systems.

Answer RQ3: MPI-CNC significantly improves acquisition frequency and efficiency by reverse engineering dedicated acquisition protocols and customized packet acquisition, reducing CNC network resource usage and increasing machining time by only 0.076%. MPI-CNC does not affect normal machining.

VI. Related Work

Cyber-attacks against CPMS directly affect production efficiency and even threaten the safety of human life. Therefore, IDS research in the field of CPMS has become an academic hotspot. Cyber-attacks on CPMS mainly focus on controlling and disrupting the manufacturing process, which is the key concern of CPMS IDS. By analyzing research results in this

field over the past few years, we have classified the state-of-the-art CPMS IDS into three categories based on detection methods.

CPMS IDS Based on Digital Twins: Digital twin technology utilizes historical data to fit a control model that simulates physical processes [32]. Balta et al. [7] collected and analyzed historical machining data from a 3-D printer to construct controller digital twin models for the 3-D printer's CNC system. By comparing the consistency between the simulated machining state of the controller digital twin model and the actual machining state, they detected cyber-attacks that tampered with the temperature parameters of the 3-D printer's nozzle heaters.

CPMS IDS Based on Side-Channel Analysis: Manufacturing equipment generates a large amount of measurement channel data during the machining process, which can indirectly reflect the machining state. Detection methods based on side-channel analysis are a popular approach in the CPMS IDS field. Chhetri et al. [4] proposed for the first time the use of audio data around manufacturing equipment to train detection models for detecting machining path tampering attacks. Bayens et al. [33] combined analysis of acoustic features of machining equipment, machining location features, and production waste features to verify product consistency. Belikovetsky et al. [5] analyzed audio data from 3-D printer stepper motors and evaluated the similarity between their audio features and audio fingerprints to detect the 3-D printing process. Mamun et al. [3] detecting 3-D printer processing trajectory changes using video stream analysis. Yoginath et al. [2] analyzed the current values of 3-D printer power lines using the Bayesian model to detect creep attacks. Shi et al. [6] extracted features from side-channel data collected by vibration sensors based on the LSTM-autoencoder algorithm and later used the OCSVM classification algorithm for anomaly detection.

CPMS IDS Based on Machining Code Analysis: The ISO has developed the ISO-6983-1 [29] standard as an international standard for NC programming languages. This standard specifies the lexical and syntax rules of NC code, forming a programming language composed of G codes and M codes. NC code contains the most complete and comprehensive control information of the machining process, and the CNC system automatically controls the machining process according to the instructions in the NC code. By analyzing the NC code, anomalies can be effectively detected. Beckwith et al. [8] extracted statistical features from NC code, including the number of G codes and M codes, as well as the frequency of *XYZ* values. They trained a machine learning anomaly classification model and conducted an offline analysis of NC code to identify anomalies. Tsoutsos et al. [34] reverse-engineered NC code to generate 3-D models, and then simulated pressure tests on these models. They discovered vulnerabilities in the NC code during this process.

VII. Discussion

The intrusion detection method proposed in this article has the following limitations.

1) The method may have difficulty in dealing with man-in-the-middle attacks implemented through tampering with the firmware of the CNC system. Such attacks require high-technical skills from the attackers and can effectively bypass the intrusion detection method proposed in this article.
2) The method is effective for application in 2-axis and 3-axis CNC machines, but it may not be able to generate detection rules specifically for 5-axis machine centers.
3) The active detection approach proposed in this article may not be applicable to CNC systems with interface authentication mechanisms. However, it should be noted that currently, most CNC systems do not restrict remote access to machining status information.
4) Low-interference acquisition methods can be applied to CNCs from different vendors, but protocol reversal demands high-technical skill. Automated protocol reversal is a meaningful task that needs to be addressed.

VIII. Conclusion

This article proposes a novel approach for runtime detection of CPMS cyber-attacks, denoted as MPI-CNC. We implement a prototype system on the FANUC CNC machine tools as an example. Specifically, MPI-CNC automatically analyzes NC programs, extracts machining process invariants, and generates attack detection rules, including machining process rules and key parameter rules. Then, using low-interference request packets, MPI-CNC actively communicates with the CNC system to collect process status and key parameters, while setting detection windows and thresholds to detect attack behaviors. In the end, we evaluate MPI-CNC in real machining scenarios using a FANUC CNC machine tool. Experimental results demonstrate that MPI-CNC exhibits excellent adaptability performance, being able to accurately detect various cyber-attacks without affecting the normal operation of the CNC system. Compared with other state-of-the-art detection models, our approach shows superior adaptability performance and detection performance.

References

[1] A. Kusiak, "Smart manufacturing," *Int. J. Prod. Res.*, vol. 56, nos. 1-2, pp. 508–517, 2018.
[2] S. Yoginath et al., "Stealthy Cyber anomaly detection on large noisy multi-material 3-D printer datasets using probabilistic models," in *Proc. ACM CCS Workshop Addit. Manuf. Secur.*, New York, NY, USA, 2022, pp. 25–38.
[3] A. A. Mamun, C. Liu, C. Kan, and W. Tian, "Securing cyber–physical additive manufacturing systems by in-situ process authentication using streamline video analysis," *J. Manuf. Syst.*, vol. 62, pp. 429–440, Jan. 2022.
[4] S. R. Chhetri, A. Canedo, and M. A. Al Faruque, "KCAD: Kinetic Cyber-attack detection method for cyber–physical additive manufacturing systems," in *Proc. IEEE/ACM Int. Conf. Comput. Aided Design (ICCAD)*, 2016, pp. 1–8.
[5] S. Belikovetsky, Y. A. Solewicz, M. Yampolskiy, J. Toh, and Y. Elovici, "Digital audio signature for 3-D printing integrity," *IEEE Trans. Inf. Forensics Security*, vol. 14, pp. 1127–1141, 2019.
[6] Z. Shi, A. A. Mamun, C. Kan, W. Tian, and C. Liu, "An LSTM-autoencoder based online side channel monitoring approach for cyber–physical attack detection in additive manufacturing," *J. Intell. Manuf.*, vol. 34, no. 4, pp. 1815–1831, Apr. 2023.

[7] E. C. Balta, M. Pease, J. Moyne, K. Barton, and D. M. Tilbury, "Digital twin-based cyber-attack detection framework for cyber–physical manufacturing systems," *IEEE Trans. Autom. Sci. Eng.*, early access, May 25, 2023, doi: 10.1109/TASE.2023.3243147.
[8] C. Beckwith et al., "Needle in a haystack: Detecting subtle malicious edits to additive manufacturing G-code files," *IEEE Embed. Syst. Lett.*, vol. 14, no. 3, pp. 111–114, Sep. 2022.
[9] H. Pearce, K. Yanamandra, N. Gupta, and R. Karri, "FLAW3D: A trojan-based cyber attack on the physical outcomes of additive manufacturing," *IEEE/ASME Trans. Mechatron.*, vol. 27, no. 6, pp. 5361–5370, Dec. 2022.
[10] M. Yampolskiy, L. Graves, J. Gatlin, J. T. McDonald, and M. Yung, "Crypto-steganographic validity for additive manufacturing (3D printing) design files," in *Proc. Int. Conf. Inf. Secur.*, 2022, pp. 40–52.
[11] T. Le et al., "Physical logic bombs in 3-D printers via emerging 4-D techniques," in *Proc. 37th Annu. Comput. Security Appl. Conf.*, New York, NY, USA, 2021, pp. 732–747.
[12] M. McCormack, S. Chandrasekaran, G. Liu, T. Yu, S. DeVincent Wolf, and V. Sekar, "Security analysis of networked 3-D printers," in *Proc. IEEE Security Privacy Workshops (SPW)*, 2020, pp. 118–125.
[13] B. Shadow. "Industrial security exploitation framework." 2020. [Online]. Available: https://github.com/w3h/isf
[14] R. R. Maiti, C. H. Yoong, V. R. Palleti, A. Silva, and C. M. Poskitt, "Mitigating adversarial attacks on data-driven invariant checkers for cyber–physical systems," *IEEE Trans. Depend. Secure Comput.*, vol. 20, no. 4, pp. 3378–3391, Jul./Aug. 2023.
[15] J. Liu et al., "ShadowPLCs: A novel scheme for remote detection of industrial process control attacks," *IEEE Trans. Depend. Secure Comput.*, vol. 19, no. 3, pp. 2054–2069, Jun. 2022.
[16] P. Mahesh et al., "A survey of cybersecurity of digital manufacturing," *Proc. IEEE*, vol. 109, no. 4, pp. 495–516, Apr. 2021.
[17] Y. Pan et al., "Taxonomies for reasoning about cyber–physical attacks in IoT-based manufacturing systems," *Int. J. Interact. Multimedia Artif. Intell.*, vol. 4, no. 3, pp. 45–54, Jul. 2017.
[18] T. Lin et al., "Susceptibility to spear-Phishing emails: Effects of Internet user demographics and email content," *ACM Trans. Comput. Human Interact.*, vol. 26, no. 5, pp. 1–28, Jul. 2019.
[19] N. Karsten and L. Jakob, "BadUSB—On accessories that turn evil," presented at Blackhat Conf., Las Vegas, NV, USA, 2014, pp. 1–28.
[20] M. Conti, N. Dragoni, and V. Lesyk, "A survey of man in the middle attacks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 2027–2051, Mar. 2016.
[21] S. B. Moore, W. B. Glisson, and M. Yampolskiy, "Implications of malicious 3-D printer firmware," in *Proc. 50th Hawaii Int. Conf. Syst. Sci.*, 2017, pp. 1–10.
[22] Y. Liu, Y. Peng, B. Wang, S. Yao, and Z. Liu, "Review on cyber–physical systems," *IEEE/CAA J. Automatica Sinica*, vol. 4, no. 1, pp. 27–40, Jan. 2017.
[23] G. Y. Dayanıklı, S. Sinha, D. Muniraj, R. M. Gerdes, M. Farhood, and M. Mina, "Physical-layer attacks against pulse width modulation-controlled actuators," in *Proc. 31st USENIX Secur. Symp.*, Boston, MA, USA, 2022, pp. 953–970.
[24] J. Gatlin et al., "Encryption is futile: Reconstructing 3D-printed models using the power side-channel," in *Proc. 24th Int. Symp. Res. Attacks, Intrusions Defenses*, New York, NY, USA, 2021, pp. 135–147.
[25] L. J. Wells, J. A. Camelio, C. B. Williams, and J. White, "Cyber–physical security challenges in manufacturing systems," *Manuf. Lett.*, vol. 2, no. 2, pp. 74–77, 2014.
[26] T. Zinner, G. Parker, N. Shamsaei, W. King, and M. Yampolskiy, "Spooky manufacturing: Probabilistic sabotage attack in metal AM using shielding gas flow control," in *Proc. ACM CCS Workshop Additive Manuf. (3D Printing) Security*, New York, NY, USA, 2022, pp. 15–24.
[27] S. R. Chhetri, A. Barua, S. Faezi, F. Regazzoni, A. Canedo, and M. A. Al Faruque, "Tool of spies: Leaking your IP by altering the 3-D printer compiler," *IEEE Trans. Depend. Secure Comput.*, vol. 18, no. 2, pp. 667–678, Apr. 2021.
[28] H. Choi et al., "Detecting attacks against robotic vehicles: A control invariant approach," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, New York, NY, USA, 2018, pp. 801–816.
[29] *Automation Systems and Integration—Numerical Control of Machines—Program Format and Definitions of Address Words—Part-1: Data Format for Positioning, Line Motion and Contouring Control Systems, International Organization for Standardization*, ISO Standard 6983-1:2009, 2009.
[30] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 26, no. 1, pp. 43–49, Feb. 1978.
[31] E. Fix and J. L. Hodges, "Discriminatory analysis. nonparametric discrimination: Consistency properties," *Int. Statist. Rev./Revue Internationale de Statistique*, vol. 57, no. 3, pp. 238–247, 1989.
[32] R. Minerva, G. M. Lee, and N. Crespi, "Digital twin in the IoT context: A survey on technical features, scenarios, and architectural models," *Proc. IEEE*, vol. 108, no. 10, pp. 1785–1824, Oct. 2020.
[33] C. Bayens, T. Le, L. Garcia, R. Beyah, M. Javanmard, and S. Zonouz, "See no evil, hear no evil, feel no evil, print no evil? malicious fill patterns detection in additive manufacturing," in *Proc. 26th USENIX Secur. Symp.*, Vancouver, BC, Canada, 2017, pp. 1181–1198.
[34] N. G. Tsoutsos, H. Gamil, and M. Maniatakos, "Secure 3-D printing: Reconstructing and validating solid geometries using Toolpath reverse engineering," in *Proc. 3rd ACM Workshop Cyber Phys. Syst. Secur.*, New York, NY, USA, 2017, pp. 15–20.

Zedong Li is currently pursuing the Ph.D. degree with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China.

His research interests include cyber–physical manufacturing systems security and intrusion detection.

Xin Chen (Member, IEEE) is currently pursuing the Ph.D. degree with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China.

His research interests include cyber–physical manufacturing systems security and intrusion detection.

Yuqi Chen received the B.Sc. degree in computer science from South China University of Technology, Guangzhou, China, in 2015, and the Ph.D. degree from Singapore University of Technology and Design, Singapore, in 2019.

He is an Assistant Professor with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China. Before joining ShanghaiTech, he was a Research Scientist with the System Analysis and Verification Group, Singapore Management University, Singapore. He employs a range of techniques, including testing, reverse engineering, program analysis, and formal methods, to develop practical solutions for securing critical cyber–physical systems. His research interests lie at the intersection of software engineering and security.

Shijie Li is currently pursuing the Ph.D. degree with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China.

His research interests include industrial control system security and intrusion detection.

Hangyu Wang is currently pursuing the Ph.D. degree with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China.

His research interests include industrial control system security and access control.

Shichao Lv received the M.S. degree in cryptographyăfrom the University of Electronic Science and Technology of China, Chengdu, China, in 2012, and the Ph.D. degree in information security from the University of Chinese Academy of Sciences, Beijing, China, in 2018.

He is a Ph.D. Professorate Senior Engineer and an M.S. Supervisor from the Institute of Information Engineering, Chinese Academy of Sciences, Beijing. His main research interests include Internet of Things security and industrial control system security.

Limin Sun received the Ph.D. degree from the National University of Defense Technology, Changsha, China.

He is currently a Professor with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China. His main research interests include Internet of Things security and industrial control system security.

Prof. Sun is also the Secretary General of the Select Committee of CWSN and the Director of the Beijing Key Laboratory of IoT Information Security Technology. He is an Editor of the *Journal of Computer Science* and the *Journal of Computer Applications*.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2019.2907247, IEEE Transactions on Services Computing

Cybersecurity in Big Data Era: From Securing Big Data to Data-Driven Security

Danda B. Rawat, *Senior Member, IEEE*, Ronald Doku and Moses Garuba

***Abstract*—"Knowledge is power" is an old adage that has been found to be true in today's information age. Knowledge is derived from having access to information. The ability to gather information from large volumes of data has become an issue of relative importance. Big Data Analytics (BDA) is the term coined by researchers to describe the art of processing, storing and gathering large amounts of data for future examination. Data is being produced at an alarming rate. The rapid growth of the Internet, Internet of Things (IoT) and other technological advances are the main culprits behind this sustained growth. The data generated is a reflection of the environment it is produced out of, thus we can use the data we get out of systems to figure out the inner workings of that system. This has become an important feature in cybersecurity where the goal is to protect assets. Furthermore, the growing value of data has made big data a high value target. In this paper, we explore recent research works in cybersecurity in relation to big data. We highlight how big data is protected and how big data can also be used as a tool for cybersecurity. We summarize recent works in the form of tables and have presented trends, open research challenges and problems. With this paper, readers can have a more thorough understanding of cybersecurity in the big data era, as well as research trends and open challenges in this active research area.**

***Index Terms*—Big Data Security, Big Data Driven Security, IDS/IPS, Data Analytics.**

I. Introduction and Background

Over the past 15 years, data has increased exponentially in various applications which has led to the big data era (Fig. 1). It is worth noting that big data has some peculiar features which can be leveraged for various purposes (Fig. 2). One of these is the use of big data for detecting risks or attacks. "As our technological powers increase, the side effects and potential hazards also escalate" is a quote by Alvin Toffler which perfectly sums up the world we live in now. Hacking was at first akin to public defacements of things. Hackers hacked for fun and for notoriety. However, these days, attacks are more calculated and motivated. Nations are accusing each other of hacking. There is also a significant rise in industrial espionage which can either be from nation-state or competing entities trying to gather information or to take away a competitor's edge as to increase their own. Additionally, we are seeing this across industries from health care to retail to government to education to the financial sector. Thus, with this much susceptibility and hacking advancements, cybersecurity has become an important field in computer science. Cybersecurity aims at reducing the attack vectors/points to a minimal, because it is impossible secure every attack point. An attacker only has to be successful once which has consequently made the job of securing systems very challenging. The number of attackers out there out-number the people trying to protect it. This is because there is so much information out there that can turn anyone into an attacker. With this in mind, cybersecurity has now gone beyond the traditional way of only focusing on prevention to a more sophisticated PDR paradigm which is: Prevent, Detect and Respond (PDR). Big data is expected to play a major role in this emerging PDR paradigm.

Big data is now a common slogan used to mean the generation of large volumes of data. Enormous amount of data are being generated at an alarming rate. This is due to the growth of the Internet. Laney [1] came up with the term the three V's which he associated with big data. These terms were volume, velocity, and variety. In addition to 3 V's, there is fourth V which is veracity. Volume represents the fact that the data being generated is enormous, velocity represents the fact that data is being generated at an alarming rate, and

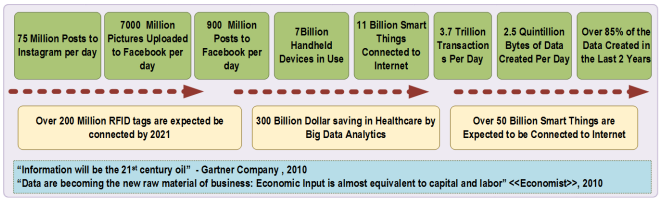

Figure 1. Big data is increasing exponentially making security harder.

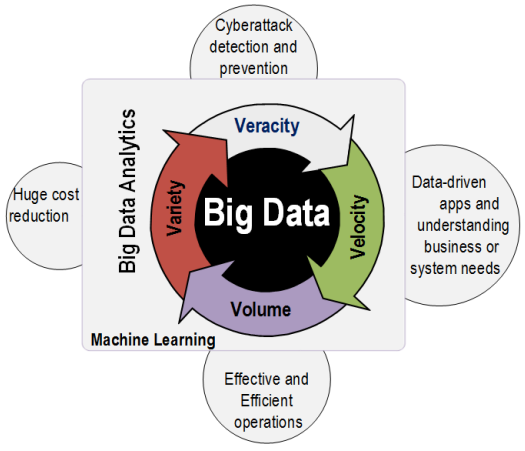

Figure 2. Big data offers typical benefits to business such as informed decisions, competitive advantages and data-driven cybersecurity.

Manuscript received 2018.

Authors are with the Data Science and Cybersecurity Center (DSC^2), Department of Electrical Engineering and Computer Science at Howard University, Washington, DC, USA. Corresponding e-mail: db.rawat@ieee.org

This work is supported in part by the U.S. National Science Foundation (NSF) under grants CNS-1658972, CNS-1650831 and and HRD 1828811. However, any opinion, finding, and conclusions or recommendations expressed in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the NSF.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2019.2907247, IEEE Transactions on Services Computing

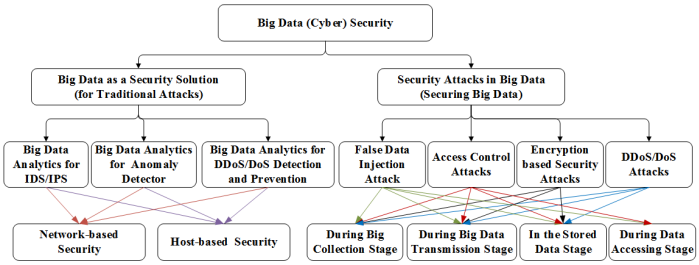

Figure 3. Big data (analytics) as a security solution and security attacks that are unique to big data in a typical big data enabled systems.

variety represents the fact that the data being generated comes in all types of forms. Big Data could be explained simply as data at rest according to Miloslavskaya et al [2]. They also highlighted the difference between big data, data lake, and fast data. Data lake holds a large amount of raw data in its original format. Fast data can be time sensitive data which may either be structured or unstructured, which is usually acted upon right away.

We have more and more data coming, and they are moving from terabytes to petabytes, which are becoming unfamiliar realms [3]. Thus, we need to find new ways of accommodating this data, and there is the need to develop models and algorithms that will enable us to work on these data, to gain insights from it. This is where Big Data Analytics (BDA) comes in. This paper explores research work done on big data enabled security and securing big data (which are categorically presented in Fig. 3).

Although there are related survey papers [4]–[16] on big data security (further details, please refer to Section IV), we present more up to date approaches, insights, perspectives and recent trends on the rapidly advancing research field of big data in the cybersecurity domain. Our approach to this covers the research work done on how big data is used as a security tool and the emergence of big data as high value asset resulting in research work done on how to secure big data. Specifically, the main contributions of this paper include:

- Presenting a comprehensive study on security aspects of big data by categorizing it into two parts: security using big data and big data driven security.
- Presenting a summary of attacks and countermeasures for big data in a tabular form for a side-by-side comparison.
- Presenting a discussion of research challenges, recent trends, insights and open problems for big data in cybersecurity.

The remainder of this paper is organized as follows. We first classify our work into two major sections (Sections II and III). We provide a comprehensive study of security using big data as well as securing big data. For each category, we present the related recent state-of-the-art literature for the different approaches. Section II focuses on the use of big data as a security mechanism. Section III tackles how big data is being protected. Section IV presents relevant survey papers along the line of this paper and the distinction of this paper from the rest of the surveys. Section V presents some research challenges and future directions in this area. Finally, we summarize the paper in section VI.

II. Security using Big Data

Top security companies joined forces to share information with each other in an attempt to gather intelligence from the shared data (SecIntel Exchange). Their goal was to provide reliable security tools for their clients, and to achieve that, they had to learn as much as possible from evolving threats that were developed each day. They understood the power of collaboration for the greater good. This was needed because with the rise of polymorphic malware and other evolving threats, they needed a lot information on these threats in order to fully understand what they were dealing with and how to counteract against it. The traditional approaches of classifying malware were proving to be futile. SecIntel Exchange data provided them with the opportunity to derive actionable insights from voluminous data. Human analysis and traditional methods such as database storage could however not keep up with the pace of the data that was being generated [17]. There was the need to adopt modern approaches. As seen in a case study conducted by Zions Bancorporation [18], it would take their traditional Security Information and Event Management(SIEM) systems between 20 minutes to an hour to query a month's worth of security data. However, when using tools with Hadoop technology, it would only take about one minute to achieve the same results. As such BDA has become an important tool in cybersecurity. Several studies have shown that the traditional approaches and human analysts can not keep up with the big data. BDA is one of the best solutions to combat these issues.

A. Big Data Analytics (BDA) as a Tool to Combat Diverse Attacks

Aypical attacks that can be subdued using big data analytics are depicted in Fig. 4). "If we know the enemy and ourselves, we need not fear the result of a hundred battles" is an excerpt from the Art of War written by the famous Chinese general, Sun Tzu. In other words, it may not be possible to know enough about our enemy, but it is definitely possible to know all that we can about ourselves and the assets we protect. To do that, we have to gather facts about the asset. This is made possible by the data it generates. This data needs to be analyzed and insights need to be drawn. BDA can help prepare, clean, and query heterogeneous data with incomplete and/or noisy records [19], something that would be hard for humans to do. Analyzing data tends to be hard when the data is heterogeneous as [20] discovered. In their work, they presented a platform targeted at achieving real time detection and visualization of cyber threats which they called OwlSight. The platform had several building blocks (data sources, big data analytics, web services and visualization) and had the ability to collect large amounts of information from a variety of sources, analyze the data and output the

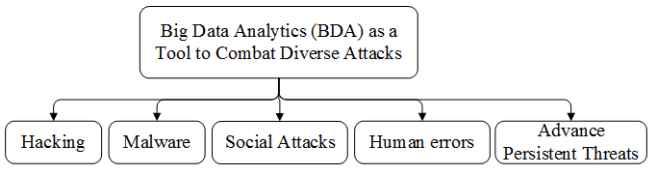

Figure 4. Big data (analytics) can combat diverse attacks.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2019.2907247, IEEE Transactions on Services Computing

findings on insightful dashboards. They did face some issues with the heterogeneity of the data. However, for machines to do the work effectively, they need to have some form of human element. Understanding a problem is half the problem solved. The authors in [21] understood this and addressed this issue by coming up with an approach that merged big data analytics with semantic methods with the aim of trying to gain further insights on the heterogeneous data by understanding it semantically. BDA can be used to gather insights making it an essential tool in cybersecurity. However, the features of big data (four V's) also make deriving insights a hard task to accomplish.

In the 2017 Data Breach Investigations Report done by Verizon, it was reported that attacks tend to come from different sources. 62% of the attacks involved hacking, 51% used malware and 43% were social attacks. 14% were a result of human errors. As such, the attacker sometimes relies on human factor in order to execute a successful attack. In such scenarios, people instead of technology become the target of an attack. Email scams and phishing are the most common form of these attacks. In a recent study [22], 52% of successful email attacks get their victims to click within an hour and 30% within 10 minutes. The authors in [23] looked into the role of big data in such attack scenarios. To gain further insights, the authors conducted two studies. The first study involved the Enron email dataset. The second study was carried out on undergraduate students to observe how email phishing broke security systems based on user behaviours. The collected data was then analyzed using Enronic software which was followed by the categorization of email topics. The authors found that, phishers or attackers can understand the behavior of email users using big data analytics, and therefore are able to generate phishing emails that created security threats based on the insights they gathered. The authors planned on proposing a framework for addressing security threat in email communication in the future. In another work, a big data enabled framework was proposed in [24] with the aim of defending against spam and phishing emails by using a global honeynet. Their framework collected data from different sources such as pcap files, logs from a honeynet, black listed sites and social networks for analysis. The framework used Hadoop and Spark for the processing of the collected heterogeneous data which was stored in Hadoop Distributed File System (HDFS). However, this framework does not provide real-time analysis for big data.

Another form of attack is Advanced Persistent Threats (APT) which are sophisticated, well-planned attacks [25]. APTs are very hard to detect, and the challenge of detecting and preventing advanced persistent threats may be answered by using big data analysis. These techniques could play a key role in helping detect threats at an early stage, especially when it uses a sophisticated pattern analysis, that works on different heterogeneous data sources. Given the numerous number of APT attacks that organizations face today, an APT security protective framework has been presented in [26]. The proposed framework integrates deep and 3D defense strategies. To protect against APT attacks, the system classifies data based on the level of confidential data. Botnet attacks is also another area where big data and machine learning techniques are deployed in. The work in [27] studied techniques for mitigating botnet attacks by using big data Analytics. The Advanced Cyber Defense Center (ACDC) orchestrated the sharing of gathered cybersecurity information on botnet attacks with the aim of defending through botnets. The work [28] proposed an architecture to address the current issue of botnet detection. They explored the possibility of employing a Self Organizing Map as an unsupervised learning approach to label unknown traffic. Financial sector is another area where big data analytics is used to prevent malicious actions or cyber attacks. The work in [29] studied using data fusion and visualization techniques in Network Forensic Analysis. Also, Cybersecurity Insurance (CI) is becoming more popular because of the increase of loss mitigation for cyber incidents for financial firms. Big data has now been employed in cybersecurity insurance, and the work in [30] proposed a framework which uses a big data approach in CI to analyze cyber incidents to gain insights in order to make better strategic decisions based on the information gathered. [31] investigated privacy and security issues associated with the sharing of financial data between institutions.

The work in [32] studied a novel Network Functions Virtualization-based (NFV) cybersecurity framework for providing security-as-a-Service in an evolved telco environment. The framework is known as SHIELD. This framework leverages BDA for detecting and mitigating threats in real time. [33] studied the idea of the construction of security monitoring systems for Internet of Things, which is based on parallel processing of data using the Hadoop platform. The proposed systems architecture has different components for the collection of data, storage of data, normalization and analysis, and visualization of data. Storage of data is done on Hadoop to improve the reliability and efficiency of processing of data requests. The work in [34] proposed a Security Information Management (SIM) enhancements using BDA. They devised a blueprint for a big data enhanced SIM, and field tested it using real network security logs. The work in [35] proposed a big data analytics model for protecting virtualized infrastructure in cloud computing. A Hadoop Distributed File system was used for the collection and storage of network logs and application logs from a guest virtual machine. Attack features were then extracted using graph-based event correlation and MapReduce parser identification of the potential paths of attack. A two-step machine learning algorithm using logistic regression and belief propagation were then applied to determine the presence of attacks. SIEM is an important tool in cybersecurity information analytics and a good source of data. The tool developed in [36] analyzes big data (gotten from SIEM) of a Fortune 500 company in order to gain insights about security threats through anomaly detection. They highlight the importance of graph analytics when it comes to intuitively understanding of business needs. Based on this, they apply graph analysis in anomaly detection by adding additional important capabilities of existing tools to their new tool, and then to visualize the network ins and outs. Finally, another use case of big data for security reasons involves a method for analyzing the security of RC4 [37]. Since attacks are diverse and come in multiple

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2019.2907247, IEEE Transactions on Services Computing

forms, BDA has been used as a cybersecurity tool to mitigate those attacks.

An area in cybersecurity where big data is used a lot is in Intrusion detection and prevention systems (IDS/IPS) research. Intrusion attempts are done to usually access information, interfere with the information or to tamper with a system thus making it unreliable and unusable. The IDS concept has been around for two decades but has recently seen a dramatic rise in the popularity and incorporation into the overall information security infrastructure [38]. IDSs are used to determine if there has been a breach or an interference in the network [39]. An IDS is often regarded as a second-line security solution after authentication, firewall, cryptography, and authorization techniques. Similarly, IPS can be classified into two categories: Network-based IPS and Host-based IPS. In network intrusion, prediction and detection is time sensitive, and needs highly efficient big data technologies to deal with problems on the fly [40]. This ensures a proactive rather than a reactive approach to cybersecurity. [41] approached this problem by developing a Proactive Cybersecurity (PCS) system. The PCS is a layered modular platform that makes use of big data collection and processing techniques to a wide variety of unstructured data to identify and thwart cybersecurity attacks. The PCS has a Targeted Vulnerability Predication (TVP) subsystem for detecting threats. Additionally, the model makes use of an Architectural Vulnerability Detection (AVD) subsystem and a risk analysis and recommender (RAR) subsystem for aiding identification and analysis of the identified risks (e.g. [16]). The work in [42] also proposed an architecture that handles IDS/IPS issues in a network. Their architecture stores and manages data from heterogeneous sources and also tries to find insights in the data. DNS data, NetFlow records, HTTP traffic and honeypot data were used in the research. Their approach however only provides offline analysis. Yang [43] presented an alternate approach that detects network anomaly at per-flow level rather than the usual per packet level which tends to bring scalability issues. They build a meta model for a number of machine learning and data mining algorithms. [44] also proposed a network security and anomaly detection framework for the big data systems for Network Traffic Monitoring and Analysis (NTMA) applications. Their framework is known as Big-DAMA. Big-DAMA is a very flexible Big Data Analytics framework (BDAF) that can perform analysis and storage of huge amounts of both heterogeneous structured and unstructured data. Big-DAMA also has batch and stream processing capabilities. Additionally, Big-DAMA utilizes Apache Spark Streaming for stream based analysis and for batch analysis, it uses Spark. For query and storage, it uses Apache Cassandra. Several machine learning algorithms are implemented by Big-DAMA for anomaly detection and network security. Big-DAMA was applied to various network attacks and anomaly detection. It was found to have the ability to speed computations by a factor of 10 in comparison to Apache Spark cluster. Security monitoring using big data has also been extended to other avenues. The work done in [45] also propose a Machine Learning model for Network-based Intrusion Detection Systems in order to detect the network security threats. Different types of ML classifiers are built using data-sets containing the labeled instances of network traffic. The focus of this research was to detect Android threats and give awareness and popularity to the users. This model can be integrated with traditional detection systems to detect advanced threats and reduce false positives. Thus, machine learning models are an essential part of BDA and have especially been used extensively in network anomaly detection.

B. Machine Learning (ML) in Cybersecurity

BDA and machine learning models go hand in hand. To provide security by deriving actionable insights, ML algorithms are needed to learn from the data. ML algorithms fall broadly into three categories: supervised learning, unsupervised learning and semi-supervised learning (which is a combination of supervised and unsupervised learning). The primary differentiator between supervised and unsupervised learning lies in the nature of the data that each uses. Unsupervised learning algorithms are used on data in which the outcome of each training sample is not known. A classic example is in malware detection. To achieve this, we extract the features from malware dataset and find groupings or similarities of the malware. The model uses the features of the data set to find its own groupings. Techniques that are used for unsupervised learning malware analysis are usually clustering algorithms and Principal Components Analysis (PCA). Supervised learning algorithms are trained on data in which the outcome of each training sample is already known. Some techniques used for doing supervised learning are linear and logistic regression, support vector machines, random forests and neural networks which are have commonly been re-branded as deep learning. Deep learning algorithms are very useful for analyzing large amounts of unsupervised data with high variety, which gives it potential in analyzing network data for intrusion detection, especially when it comes to NIDS [46], [47]. [48] tackled this issue when they used a deep learning technique called Self-taught Learning(STL) on the NSL-KDD dataset for intrusion detection on a network.

However, deep learning has some challenges in big data [49]. Its adaptability can be used as a vulnerability when attackers exploit the Machine Learning models. Adversarial examples [50] are machine learning inputs specifically designed to trick the ML model into producing a different output. Various works that have been done on this area try to refine the models [51]. [52] however propose a different approach to detecting adversarial examples. This approach is called feature squeezing which involves the reduction of the search space available to an adversary by merging samples that correspond to many different feature vectors in the original space into a single sample. With the advancement of Generative Adversarial Networks and big data, attackers are using artificial intelligence to circumvent some of the machine-learning automated processes. In lieu of this, a more effective approach is the merging of human and machine elements. Vimod [53] proposed an approach were humans and machine collaborate together. They used high-functioning autistic graduates with specific attributes to monitor networks and network flows.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2019.2907247, IEEE Transactions on Services Computing

The other work [54] that incorporated the use of big data to assist humans studied data triage, and how helpful it is in identifying true attack patterns in a noisy data. This approach tries to automatically generate data triage automaton by tracing the actions of security analysts. This approach is different from existing data triage automaton like SIEM, because unlike SIEM, which requires analysts to manually generate event correlation rules, their approach mines data triage rules out of cybersecurity analysts' operation traces. It can be sheen that attackers are using artificial intelligence to trick ML models. Human and machine working together is one of the effective ways to combat these attacks.

III. Securing Big Data

Previous section presented how security can be achieved with big data. This section presents how to secure big data against different attacks. Typical techniques for securing big data are shown in Fig. 5. When data gets really big, securing it becomes really difficult. In [58], authors studied the security issues associated with big data and cloud computing. They identified the fact that most organizations outsource database in the form of big data in to the cloud. Cloud computing however still has many risks associated with it. The goal in [58] was to find security vulnerabilities in the cloud in order to inform vendors about recent vulnerabilities. They noted that confidentiality, integrity and availability in that order as the most important security issues a cloud provider faces. Confidentiality in this scenario would mean the protection of data against unauthorized interference or usage. Integrity would be the prevention of unauthorized and improper data modification. Availability would be akin to data recovery from hardware, software and system errors, and also from data access denials. However, confidentiality is the most important aspect when it comes to big data protection. Several data confidentiality techniques exist with the most notable ones being access control and encryption.

A. Access Control and Encryption Techniques for Big Data

Encryption and access control are similar in the sense that they are both synonymous with privacy and prevention. A notable difference however is that, encryption usually deals with the confidentiality of data. Data can be available to either a trusted or untrusted entity. Encryption ensures that only authorized trusted entities can view the data. Access control however tries to limit the access to data. The data limitations usually happens amongst trusted parties. For this reason, encryption techniques have to be stronger than access control techniques. Encryption imposes very strong limitations over data confidentiality. However, encryption is not an easy task. It tends to be computationally expensive and it has

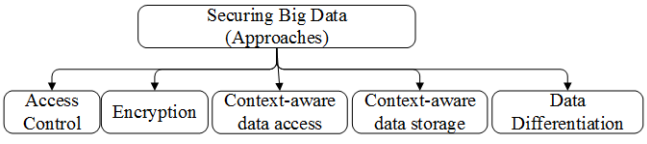

Figure 5. Securing the big data.

scalability issues (many users requiring access to the same data). Access control tends to be more flexible, and is easier to implement. When Big Data is transmitted to the cloud, a security issue emerges. Most organizations would not want their data in the hands of another organization, thus the need for encryption. A common approach is the use of data masking schemes. When the data is transmitted, it is not encrypted because the approaches used to transmit the data requires that the data be decrypted. This exposes the data to attacks. Confidentiality breach is the biggest threat to big data thus the encryption could be used as the primary big data protection technique.

In [59], authors studied the data transmission issues. They proposed computing on masked data to solve this. They proposed an incremental work to improve upon the already existing Fully Homomorphic Encryption (FHE) and other data masking techniques by decreasing the over head associated with other FHE techniques. The work in [60] also tried to improve on a Fully Homomorphic Encryption scheme for big data. It attempted to do this by reducing the public key size with the aim of making their scheme more efficient. The work in [61] proposed a model for the protection of data privacy using a fully homomorphic non-deterministic encryption. The proposed data protection model ensured the prior encryption of data before it was transmitted and therefore avoidance of the loss of data. The proposed system however only accepted numerical data. The output from the system was a result gotten from the computation of the encrypted data which is similar to that of the plaintext. In the future, the authors will look into the improvement of this anonymized data protection approach. The work in [62] explored the use of Format-Preserving Encryption (FPE) data masking scheme for voluminous data. The approach chooses various FPE algorithms depending on the type of data and what needs to be done. Spark framework was used. The authors chose FPE encryption technique because the ciphertext of FPE will still retain the original format of the plaintext instead of unreadable binary string. The ciphertext will now not contain any sensitive information. This approach however has its drawbacks. The encryption speed is slow when compared to other traditional symmetric algorithms. In one FPE method call, the algorithm calls the block cipher many times thereby making it inefficient. Another commonly used encryption is Attribute Based Encryption (ABE). The work in [63] presented a framework for fine-grained data access control to Personal Health Records (PHR) in the cloud that uses Attribute-Based Encryption as an encryption method to ensure that each patient has a unique key based on his/her attributes. The data could be accessed under multi owner settings. It was not only free of errors, but also protected the data from malicious parties aiming at deceiving the data users. In another paper involving ABE, Yang et al [64] addressed some of the shortcomings of ABE encryption in cloud data storage services. They proposed a variant of ABE which is a novel distributed, scalable and fine-grained access control scheme based on the classification attributes of the cloud storage object. Their goal was to improve on the shortcomings of ABE by taking into account the relationships among the attributes. The work in [65] investigated a hybrid approach

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2019.2907247, IEEE Transactions on Services Computing

Table I
SECURING BIG DATA

Method/reference	Goals	Source of Data	Tools/Technologies
Security threats for big data [23]	Mitigating Phishing Attacks	Enron E-mail Dataset	Enronic Software
A big data architecture for security data [24]	Defend Against Spam and Phishing	pcap files, logs from honey net	Hadoop, Spark
Data mining methods for detection of malicious executables [55]	Detect malicious malware	Malicious and benign executable binaries	Machine Learning and Data Mining Algorithms
A practical solution to improve cybersecurity [53]	Security monitoring tool	Network dataset	Data Mining Techniques, High Functioning autistic graduates
Automate Cybersecurity Data Triage [54]	Help security analysts with data triage	The operation traces of security analysts on IDS logs and Firewalls	Data modeling and mining Techniques, Humans
Analyzing and Predicting Security Event Anomalies in BDA Deployment [36]	Improve SIEM by adding important features.	Traditional SIEM systems	Data Mining, Graph Analytics
Network Information Security on Big Data [26]	Advanced Persistent Threat Detection	Network data set	Big Data Analytics, Network event collection techniques, Big Data correlation analysis
Big Data machine learning and graph analytics [56]	Combining batch and stream data processes for efficiency reasons	Hetereogeneous Big Data (any type of data)	Lambda architecture
SIM in light of Big Data [34]	Cyber attack detection	Security logs	Machine learning techniques
Data fusion & visualization [29]	Network forensic investigation	Network logs	Data fusion techniques, Visualization, Self Organizing Map
Owlsight: Platform for real-time detection and visualization of cyber threats [20]	Real time detection and visualization of threats	Heterogeneous network data	Big Data Analytics, Web services, Data visualization
Predicting and fixing vulnerabilities before they occur: a Big Data approach [41]	Proactive Defense (Prevention better than cure approach)	Heterogeneous network data	Big Data Analytics techniques, Machine Learning
Machine learning classification model [45]	Network Intrusion Detection System in Android phones	Android data	Machine Learning Algorithms
A Big Data architecture for large scale monitoring [42]	Intrusion detection and prevention systems	NetFlow records, HTTP traffic and honeypot data	Shark, Spark, Machine Learning algorithms
A Scalable Meta-Model for Big Data Security Analysis [43]	Detect network anomaly at per flow level rather than the usual per packet level which tends to bring scalability issues	Network data	Machine learning and Data Mining Algorithms
Network security and anomaly detection [44]	Intrusion Detection System	Network flow Data	Spark, Cassandra, Machine Learning Algorithms
SHIELD: A novel NFV-based cybersecurity framework [32]	Security as a Service(SecaaS) to protect applications on Software	Heterogeneous cybersecurity data	Big Data Analytics, Machine Learning
Security evaluation of RC4 using Big Data analytics [37]	Analyzing the security of RC4	RC4 Algorithm	MapReduce, Big Data Analytics
Big Data Based Security Analytics for Protecting Virtualized Infrastructures in Cloud Computing [35]	Big Data analytics model for protecting virtualized infrastructure in cloud computing	Network logs and application logs from a guest virtual machine	Machine Learning algorithms
Big Data security analysis approach using computational intelligence techniques [57]	Deduce the security status of the desktop and sources and causes of security breaches	Log file of Windows Firewall	Computational intelligence techniques
Data analytics on network traffic flows for botnet behaviour detection [28]	Issue of botnet detection	Network Traffic Data	Self Organizing Map as an unsupervised learning approach to label unknown traffic

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2019.2907247, IEEE Transactions on Services Computing

that combines symmetric cryptography and ABE to secure big data. They wanted to combine the flexibility of attribute-based cryptography and the efficiency of symmetric cryptography. They use Ciphertext-Policy Attribute-Based Encryption (CP-ABE) and AES encryption. In another form of big data encryption scheme, the work in [66] proposed an encrypted MongoDB which utilizes a homomorphic asymmetric cryptosystem which can be used for the encryption of user data and in achieving privacy protection. Thus, the FPE, FHE and ABE are the more popular researched big data encryption techniques.

A model for encrypting both symmetric and asymmetric data was presented in [67] which sought to overcome the limitation of asymmetric encryption techniques such as key exchange problem and the limited size of data and which in turn made it irrelevant for big data applications. Their proposed technique was known as BigCrypt which uses a probabilistic approach to Pretty Good Privacy Technique (PGP). BigCrypt encrypts the message with a symmetric key and encrypts the symmetric key using a public receiver key which is then attached to the message. The message is then sent. At the receiver end, the symmetric key is extracted and then asymmetrically decrypted and used for decrypting the main message. The proposed model was tested on local, web, and cloud server and was found to be efficient. Furthermore, a framework for securing the sharing of sensitive data on a big data platform was proposed in [68]. Sharing sensitive data securely reduces the cost of providing users with personalized services in addition to providing value-added data services. The proposed scheme secures the distributed data, securely delivers it, stores it, and ensures secure usage. Semi-trusted big data is also destroyed. The proposed scheme uses a proxy re-encryption algorithm that is based on heterogeneous ciphertext transformation. The scheme also utilizes a user process protection method based on a virtual machine monitor that supports other system functions. This framework ensures data security while ensuring it is shared safely and securely. Sharma and Sharma in [69] discussed the protection of big data using neural and quantum cryptography. Neural cryptography incorporates the concept of artificial neural networks with classical cryptographic algorithms while quantum cryptography makes use of the phenomenon of quantum physics for securing communications. The authors also provided a comparative analysis between quantum and neural cryptography based on the methodologies that both techniques employ. From the analysis, the authors showed that a quantum computer makes use of quantum mechanisms for computation which are very powerful and can therefore crack complicated problems such as discrete logarithmic problem in a small duration. Neural key exchange protocol is also shown not to depend on any number theory. The analysis also indicates that neural networks probably have higher protection. The work in [70] proposed an efficient group key transfer protocol necessary for ensuring secure group communication on big data. The proposal does not use an online key generation centers (KGC) which is based on 3-LSSS (Linear secret sharing scheme) in that three modular multiplications are needed. Additionally, the protocol uses Diffie-Hellman key agreement. The proposed group key transfer scheme consists of two sections; two party secret establishment section and a section for the group session key transfer. The proposed group key transfer scheme was analyzed to verify its elements of key freshness, key confidentiality, and key authentication. Furthermore, the work in [71] proposed a new encryption scheme that can be used on big data that uses double hashing instead of a single hash. Double hashing they claim eliminates the threat of known cryptanalysis attacks. The work in [72] discussed primarily about the enhancement of CAST block algorithm for the security of big data. Their contribution to the enhancement of the cast block algorithm involved the use of one S-box instead of 6, and an approach to make it more dynamic. The work in [73] presented a framework that is Light-weight Encryption using Scalable Sketching(LESS) for reducing and encrypting the processing of big data on low power platform. This contains two kernels."sketching" and "sketch-reconstruction". Orthogonal Matching Pursuit (OMP) algorithm is implemented on the domain-specific Power Efficient Nano Cluster platform that acts as a hardware accelerator and ARM CPU for big data processing. Finally, the work [74], discuss the security issues of heterogeneous, multimedia big data. They tackle resource constraint issues such as limited computation and energy resources. They proposed data encryption models that deals with this issue by reducing the computation overload on weak nodes and by replacing the current encryption models with an improved version based on SAFE encryption scheme to improve it. The work in [75] mentioned a new approach for the privacy and security protection of clinical data through the use of the art encryption scheme and attribute based authorization framework.

For the access control and privacy of big data, the work in [77] presented a hybrid approach based framework that composes and enforces privacy policies to capture privacy requirements in an access control system. Gao et al [78] presented a cloud security control mechanism based on big data. Cloud computing was observed to have increased the amount of data in the network. Due to this, big data leaks and losses occurred. Therefore, there was the need to provide the necessary level of protection. To that end, they conducted an analysis on big data, analyzed the current big data situation. Gupta et. al. [79] proposed a security compliance model for big data systems. The model provides security and access control to big data systems at the initial stage. The proposed system has four models; the library, low critical log, high critical log, and a self-assurance system. The design of this system ensures real time analysis of big data. The initial level of security provided by the model is facilitated by its web directory and its self-assuring framework that identifies and differentiates genuine users and critical users. The relationship analysis tool of the users blocks users who are deemed not to be genuine. In [76], the authors proposed a framework for privacy policy for big data security. The proposed framework makes use of different techniques including security policy manager, fragmentation approach, encryption approach, and security manager. The characteristics of the privacy policy required flexibility, integration, customizability, and context-awareness. The framework works by receiving data from the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2019.2907247, IEEE Transactions on Services Computing

Table II
RESEARCH ON ACCESS CONTROL AND ENCRYPTION TECHNIQUES ON BIG DATA

Method/reference	Problem	Solution
Computing on masked data: a high performance method for improving Big Data veracity [59]	Data not encrypted during Transmission	Improving on FHE by decreasing overhead
A faster fully homomorphic encryption scheme in Big Data [60]	Data not encrypted during Transmission	Improving on FHE by reducing public key size
A Data Masking Scheme for Sensitive Big Data Based on Format-Preserving Encryption [62]	Retain the original format of the plaintext instead of unreadable binary string during transmission	Format-Preserving Encryption (FPE) data masking scheme that chooses various FPE algorithms depending on the type of data and what needs to be done
Big Data Privacy Using Fully Homomorphic Non-Deterministic Encryption [61]	Data Security in the cloud during transmission	Fully Homomorphic non-deterministic encryption
Securing Personal Health Records in Cloud Computing: Patient-Centric and Fine-Grained Data Access Control in Multi-owner Settings [63]	Securing Personal Health Records in the cloud	Attribute-Based Encryption to ensure that each patient has a unique key based on his/her attributes
A Fine-Grained Access Control Scheme for Big Data Based on Classification Attributes [64]	Shortcomings of ABE encryption in cloud data storage services	Improve on the shortcomings of ABE by taking into account the relationships among the attributes
A digital envelope approach using attribute-based encryption for secure data exchange in IoT scenarios [65]	Improving Big Data Security	Better security by combing the flexibility of attribute-based cryptography and the efficiency of symmetric cryptography
CryptMDB: A practical encrypted MongoDB over Big Data [66]	Encryption of user data and in achieving privacy protection	Encrypted MongoDB which utilizes a homomorphic asymmetric cryptosystem
BigCrypt for Big Data encryption [67]	Overcome the limitation of asymmetric encryption techniques	BigCrypt(uses a probabilistic approach to Pretty Good Privacy Technique)
A Multi-level Intelligent Selective Encryption Control Model for Multimedia Big Data Security in Sensing System with Resource Constraint [74]	Security issues of heterogeneous, multimedia Big Data under resource constraints	Proposed a SAFE encryption scheme to replace old encryption models
Secure sensitive data sharing on a Big Data platform [68]	Securing the sharing of sensitive data on Big Data platform	Used proxy re-encryption algorithm based on heterogenous ciphertext transformation
Big Data protection via neural and quantum cryptography [69]	Protecting data	Using neural and quantum cryptography
Novel group key transfer protocol for Big Data security [70]	Secure group communication on Big Data,	Efficient group key transfer protocol using Diffie-Hellman key agreement
Double-Hashing Operation Mode for Encryption [71]	Cryptanalysis attacks	Used double hashing instead of a single hash
Enhancement CAST block algorithm to encrypt Big Data [72]	Enhancement of the cast block algorithm,	Use of one S-box instead of 6 to make it more dynamic
Less: Big Data sketching and encryption on low power platform [73]	Reducing and encrypting the processing of Big Data on low power platform	Light-weight Encryption using Scalable Sketching
Policy enforcement for Big Data security [76]	Privacy policy for Big Data security	Analyzes data, extracts the privacy policies, identifies sensitive data, then fragmentation algorithm executed on sensitive data
Managing the privacy and security of e-health data [75]	Privacy and security protection of clinical data	Art encryption scheme and attribute based authorization framework

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2019.2907247, IEEE Transactions on Services Computing

customer and then analyzing it. It is then followed by the extraction of the privacy policy and finally the identification of sensitive data. Once sensitive data has been identified, a fragmentation algorithm was executed on the sensitive data. The security modules play the role of identifying sensitive data from non-sensitive data and then regulating its access. The work in [80] proposed a privacy protection technology and control mechanism for medical big data. The proposed framework has four main phases; the setup phase, Encrypt and Upload phase, Download phase, and Share File phase. The system first de-identifies the patient personal privacy data, encrypts it using digital signature mechanisms to protect data confidentiality and the authentication of the data. The communication security of the data in the system is protected using the Diffie-Hellman session key while the integrity of the medical records is protected using a digital signature scheme. Access control is not as big as it used to be due to the evolving threats landscape but is still an important research area in big data security today.

B. Alternative Approaches to Securing Big Data

Encryption and access control were the mainstream approaches for big data security. However, researchers have tried other approaches that may or may not involve some form of encryption. The nature of big data makes it difficult to protect everything. Some researchers have tried to determine the important parts of big data to protect those parts only. The work in [81] tried to tackle the issue of securing personal health records by proposing a framework that classifies data based on a person's societal importance and determining the sensitivity levels of the data. Furthermore, [82] tried to secure the attributes of big data that are really important/valuable because protecting everything is a difficult task. They use data masking to protect these high valued attributes. To determine the attributes that are of value, they use a ranking algorithm that prioritizes attributes for big data security. Authors in [83] proposed an attribute selection method for protecting the value of big data by determining attributes that have higher relevance using a ranking algorithm, and providing security measures. In the paper [84], the authors focused on the characteristics of big data and proposed the protection of big data using a security hardening methodology that makes use of attribute relationships. The relationship between the various attributes are expressed using nodes and edges. The proposed model works by limiting the attribute to protect value. The model works by first extracting all the attributes of the targeted big data. The nodes are then arranged circularly followed by the establishment of the relationship between the nodes. The relationship is then set based on either the domain specific criteria or the universal criteria. Finally, the protecting nodes are selected followed by the determination of how to protect the selected nodes. Thus, protecting everything in big data is hard. An easier approach is to find what is important and protect that part only.

Encryption has been used with other techniques as well. The work in [85] proposed a method to secure Multimedia Big Data (MBD) in the healthcare cloud by using a Decoy Multimedia Big Data (DMBD). The DMBD uses fog computing and a pairing based cryptography that will be used to secure the MBD. Fog Computing was utilized for the storage of the decoy files. In their method, the decoy files are retrieved at the onset unlike other methods that usually waits until there is an attack before the decoy files are called. Thus, both attacker and legitimate users both see a decoy file until the legitimacy is confirmed. Aynur in [86] presented a new technique for securing big data in medical applications. The methodology combines three major techniques that include data hiding, image cryptography and steganography. These techniques facilitate safe and de-noised transmission of data. A stream cipher algorithm is used for encrypting the original image. Patient information is then embedded in the encrypted image by means of a lossless data embedding technique together with a key for hiding data to enhance the security of data. Steganography is then applied in embedded image with a private key. When the message gets to the receiver, it is decrypted using inverse methods in reverse order. Efficiently securing big data continues to become a difficult challenge because of big data's variety, volume and veracity issues. The ability to deal with space and time issues by correlating events would play an important role in securing big data. [87] discussed the growth of social media network such as Facebook and cloud computing, and how sharing of multimedia big data has become easier than ever. However, its increased used is faced with issues of piracy problems, illegal copying, and misappropriation. To address these challenges, the authors in this study proposed a system for protecting multimedia big data distribution in social networks. The scheme utilizes a Tree-Structured Harr (TSH) transform. In this scheme, a homomorphic encrypted domain for fingerprinting by means of social media network analysis is applied. The scheme aims at mapping hierarchical social networks into trees structure of the transform of TSH for coding, encrypting, and fingerprinting of JPEG2000. Finally, in [88], authors discussed the use of traditional security framework for protection of the smart grid comes with several disadvantages such as late detection of attacks when damage has already occurred. To address this problem, the authors in this study proposed a security awareness mechanism based on the analysis of big data in the smart grid. The model has three main parts which include the extraction of network security situation factors, network situational assessment and network situational prediction. The method works by integrating fuzzy cluster based analytical tools, reinforcement learning and game theory. The integration of these components facilitates security situational analysis in the smart grid. Simulation tests and experiments showed the proposed system to have high efficiency and low error rate.

Sometimes, we have to protect data from the people and the systems that interact with it. Pissanetzky [89] examined the problem of software vulnerability and the accumulation of unprocessed information in big data. According to the authors, these problems are created by human interventions. To solve these problems, the author proposed the complete elimination of human intervention. In this approach a causal set was taken as the universal language of all information and computations. Additionally, the author also proposed the confinement of the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2019.2907247, IEEE Transactions on Services Computing

use of programming languages to the human interface and therefore a creation of an inner layer of mathematical code that is expressed as a causal set. Furthermore, this paper also includes experiments and computational verifications of the theory and proposed applications of this approach to science and technology, computer intelligence, and machine learning. Also, [90] researched on how to protect both the data and the program that processes the data while taking into consideration the big data processing requirements. They propose a model that aims to address the issue by hiding operations performed using steganography and FHE in order to meet the security requirements necessary to protect outsourced data. However, the user's computation cost is somewhat high and the solution does not apply to all applications. The work in [91] addressed the use of cloud computing and how it provides an organization with various services for meeting their various needs. However, data storage in cloud computing could be accessed by cloud operators and therefore compromise information privacy and security. In this respect, this study proposed an approach for splitting and separating the stored data on distributed cloud servers and therefore prevent access by cloud operators. The proposed model was known as Security-Aware Efficient Distributed Storage (SA-EDS) and was based on two algorithms; the Efficient Data Conflation (EDCon) algorithm and Secure Efficient Data Distributions (SED2) algorithm. These algorithms were tested and proved to be efficient. The authors of [92] proposed a Field Programmable Gate Arrays (FPGA) based solution for running BLAST algorithm in a secure manner in MapReduce framework using cloud computing. The proposed system protects data from cloud service provider (CSP) through leveraging on bitstream encryption mechanism and FPGAs tamper resistant property. The authors also put into consideration the risks that arise from keys distribution and propose countermeasures for handling it. The work in [93] studied an approach that assesses the risk behind various applications and provides an explanation of the ability of the application to protect data using a specific security classification level. The proposed method has three main components; Automatic Risk assessment of the Application, Automatic Generation of Criteria for storage of specific data, and Automatic Reporting. The report facilitates the recommendation of the appropriate security level. The work in [94] proposed a hadoop system that would both secure and maintain the privacy of big data. They tried to do this by using four encryption techniques randomly. However, these encryption techniques are time consuming, thus they proposed a buffer system where the buffer stores information whilst the system works on the previous data stored in order to prevent information loss.

Knowing the characteristics of the data is an important aspect of protecting the data. Singh [95] studied the value of real-time BDA and the security challenge that comes with protecting big data. Singh notes that, proper protection of big data should focus on volume, velocity, and variety of big data. Multilevel security for big data should be provided at the application, operating systems, and network levels. However, using the traditional protection mechanism is challenging for large volumes of data that is changing continuously. For this reason, Singh recommends the use of machine learning for protection of big data with focus on supervised and unsupervised learning. Yang [96] examined the visualization of network security under the big data environment. The authors first look at the 5V characteristics of big data including volume, velocity, variety, value, and visualize. These 5V features are then mapped onto network security data followed by a description of the visualization of the data security technology. The network visualization technology proposed include the use of radial traffic analyser and SRNET. They also proposed safety visualization using ClockMap and discussed diversified technologies for visualization of big data. With the increasing volume of big data, security and privacy issues also continue to increase. Peer to peer (P2P) protocols such as BitTorrent are now being used to widen the transfer of big data. However, this increase has also attracted widespread security challenges. Research indicate that P2P are sophisticated in data transfer but experience challenges when distributing big data. Ban et. al. [97] presented a study on the early identification of attacks using the darknet. The system works by first exploring the regularities in communications from the attackers. This is achieved using an itemset mining engine. It then characterizes the activity level of each pattern of attack creating a time series. A clustering algorithm is then applied to extract the most prominent patterns of attack. The attack patterns are clustered into groups having similar activities. Visual hints on the relationship of the various attacks is then provided using a dimension reduction technique. Attacks that feature prominently are then the picked up for further analysis by experts. The authors showed that the proposed system was efficient in early attack detection.

The union of blockchain and big data will make sure that the data that is generated from the blockchain is trustworthy. This is because the provenance of the data is known. Also, the likelihood of the data being interfered with is very low. This is made possible through the blockchain's consensus mechanism and its secure cryptographic hash function which ensures data immutability. Data manipulation would require tremendous amount of hash power in order to be achieved. The centralized way of storing data is prone to data breaches and hacks [109]. This method is susceptible to single point of failure of problems as well. Distributed data storage tries to take data away from the hands of these centralized authorities, thereby taking away various security risks. The work in [106] proposed a model for security sharing based on blockchain technology to address trust issues often associated with circulation of data. The proposed model provides a credible platform for sharing data between data producers and demand parties though building a decentralized security system for the circulation of data. The security system is built using blockchain and smart contract. While blockchain technology ensures the traceability of data, the automated execution of smart contract provides the security for data security sharing. The decentralized architecture ensures the data provider does not suffer from the risks of sharing data from a centralized storage system. On the user's side, transparency in the collection of information is assured by the blockchain operation model and thereby bringing stronger user privacy protection. [110]

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2019.2907247, IEEE Transactions on Services Computing

Table III
RESEARCH ON ALTERNATIVE APPROACHES TO SECURING BIG DATA

Method/reference	Problem	Solution
A framework for providing security to Personal Healthcare Records [81]	Securing personal health records	Framework that classifies data based on societal importance and sensitivity levels
A novel data security framework using E-MOD for Big Data [82]	Securing important attributes of Big Data	Ranking algorithm to determine attributes and data masking to protect them
A Security Model for Preserving the Privacy of Medical Big Data in a Healthcare Cloud Using a Fog Computing Facility With Pairing-Based Cryptography [85]	Securing Multimedia Big Data (MBD) in the healthcare cloud	Use fog computing to store Decoy Multimedia Big Data (DMBD)
A space-and-time efficient technique for Big Data security analytics [98]	Space and time issues of Big Data	Bloom filter and its variants
Another Look at Secure Big Data Processing: Formal Framework and a Potential Approach [90]	Protecting both the data and the program that processes the data	Hiding operations performed using steganography and FHE
Attribute relationship evaluation methodology for Big Data security [83]	Attribute selection method for protecting the value of Big Data	Determining attributes that have higher relevance using a ranking algorithm
Real time Big Data analytic: Security concern and challenges with Machine Learning algorithm [95]	real time Big Data analytics and its security challenge,	use of machine learning for protection of Big Data
Research on Network Security Visualization under Big Data Environment [96]	Visualizating network security under the Big Data environment	Use of radial traffic analyser and SRNET
Secure and private management of healthcare databases for data mining [99]	Secure and private management and mining of data in health care	Executing SQL queries on encrypted data and the return differentially-different answers on the outsourced databases
Secure Distribution of Big Data Based on BitTorrent [100]	Security issues accompanying P2P Big Data transmission avenues	Scheme for secure and efficient distribution of Big Data on BitTorrent networks using bittorrent protocols
Secure multimedia Big Data sharing in social networks using fingerprinting and encryption in the JPEG2000 compressed domain [87]	Protecting multimedia Big Data distribution in social networks	Homomorphic encrypted domain for fingerprinting by means of social media network analysis
Security in Big Data of medical records [86]	Securing Big Data in medical applications	Data hiding, image cryptography and stenography
Security-aware efficient mass distributed storage approach for cloud systems in Big Data [91]	Data storage in cloud computing could be accessed by cloud operators and therefore comprise information privacy and security	Splitting and separating the stored data
Security-as-a-service in Big Data of civil aviation [101]	Data protection and privacy preserving services architecture in civil aviation	Civil aviation security data authentication through OpenSSL identity and attribute-based authorization
Towards Early Detection of Novel Attack Patterns through the Lens of a Large-Scale Darknet [97]	Early identification of attacks using the darknet	Itemset mining engine to explore regularities in attack, then machine learning algorithms (clustering) to determine attack patterns and predict attacks
Big Data analysis based security situational awareness for smart grid [88]	Disadavantage of using traditional security framework for protection of the smart grid	Security awareness mechanism based on the analysis of Big Data in the smart grid
Big Data security hardening methodology using attributes relationship [84]	Protection of Big Data using a security hardening methodology	Makes use of attribute relationships to achieve it
On the Future of Information: Reunification, Computability, Adaptation Cybersecurity, Semantics [89]	Problem of software vulnerability and the accumulation of unprocessed information in Big Data	Complete elimination of human intervention

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2019.2907247, IEEE Transactions on Services Computing

Method/reference	Goal	Solution
Privacy preserving large scale DNA read-mapping in MapReduce framework using FPGAs [92]	Running BLAST algorithm in a secure manner in MapReduce framework using cloud computing	a Field programmable gate arrays (FPGA) based solution and a bitstream encryption mechanism
Efficient privacy-preserving dot-product computation for mobile Big Data [102]	Secure privacy-preserving scheme in mobile Big Data	Privacy-preserving dot product
Privacy-Preserved Multi-Party Data Merging with Secure Equality Evaluation [103]	Merging of encrypted data	Data anonymization technique that ensures privacy in the collection and merging of data and secures multiparty sharing of data without the involvement of third parties
Proposition of a method to aid Security Classification in Cybersecurity context [93]	Managing security classification	Assessing the risk behind various applications and providing an explanation of the ability of the application to protect data using a specific security classification level
Toward a cloud-based security intelligence with Big Data processing [104]	Cloud based security intelligence system for Big Data processing	Highly scalable plugin based solution that monitors Big Data systems in real time and therefore reducing the impact of attacks or threats on a distributed infrastructure
Research about New Media Security Technology Base on Big Data Era [105]	Security threats of the new Big Data in digital era	"Blocking as loose" technology for protection, intelligent cleaning of new media Big Data, and mining of Big Data in a safe manner.
Big Data Model of Security Sharing Based on Blockchain [106]	Model for security sharing based on blockchain technology to address trust issues often associated with circulation of data	blockchain and smart contract
Big Data for Cybersecurity: Vulnerability Disclosure Trends and Dependencies [107]	effective vulnerability management for organizations dealing with Big Data	proactive Big Data vulnerability management model based on rigorous statistical models with the capability of simulating anticipated volume and dependence of vulnerability disclosures
New approach for load rebalancer, scheduler in Big Data with security mechanism in cloud environment [108]	Rebalancing and scheduling of loads in Big Data environment,	Proposed scheme uses a load balancing algorithm that merges with MD5 and DES encryption algorithm
Hadoop eco system for Big Data security and privacy [94]	Secure and maintain the privacy of Big Data	Four encryption techniques. Using a buffer system where the buffer stores information whilst the system works on the previous data stored in order to prevent information loss

proposed a system called MeDShare which is a blockchain based and provides data source auditing, and control for shared medical data in cloud repositories. The MeDShare system helps to transfer and share data from one source to another, and are recorded in a tamper-proof manner. The marriage of blockchain and Big Data is imminent as blockchain ensures data integrity.

The work in [102] proposed a secure privacy-preserving scheme using dot product in mobile big data. Privacy-preserving dot product has been used in data mining for a long time as it helps in curbing statistical analysis attacks. It is now being used in big data for its anonymous private profile matching. The paper was just an exploratory research on its use in mobile big data. There is however still room for further improvement. The work in [103] explored the idea of a data anonymization technique to support merging of encrypted data. The technique ensures the protection of privacy in the collection and merging of data and secures multi-party sharing of data without the involvement of third parties. The merging result as proposed in this study does not lead to the violation of the privacy of the individual. Additionally, the proposed mechanism allows for storage of different datasets from different parties in multiple third-party centers without leaking the identity of owners of that data. The anonymized data can be joined securely within a reasonable time. Experiments conducted by the authors indicated that 100,000 entries of data can be merged in about 1.4 seconds using the optimized secure merging procedure. To answer the question of how security classification can be managed on a system. In addition, the work in [104] proposed a cloud based security intelligence system for big data processing. The authors provide a highly scalable plug-in based solution that monitors big data systems in real time and therefore reduced the impact of attacks or threats on a distributed infrastructure. The solution proposed here was named Advanced Persistent Security Insights System (APSIS). APSIS works by taking advantage of a SIEM system including aggregation, correlation, alerting, and forensic analysis. This is exposed to big data but with security intelligence to provide accurate results. APSIS monitors all devices on the network that generate log files and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2019.2907247, IEEE Transactions on Services Computing

therefore assures security. In the future, the authors aim at exploring the proof of concept to evaluate the robustness of the proposed architecture. The work in [105] started by looking at security threats of the new multimedia heterogeneous big data. The first threat was lack of effective mechanisms for the protection of this new media ownership as DRM is facing challenges. Secondly, there is lack of a clean environment for the consumption of new media. To overcome these challenges, Lu proposed the use of "blocking as loose" technology for protection, intelligent cleaning of new media big data, and the mining of big data in a safe manner. [98] summarized how bloom filter and its variants are used to secure big data. After various experiments, they concluded that, bloom filter can be used for efficiency reasons because there are space and time issues when it comes to analyzing and indexing big data which would in turn lead to better security analytics. The research work in [99] proposed a framework for secure and private management and mining of data that addresses both security and privacy issues in health-care data management especially in outsourced databases. The solution works by executing SQL queries on encrypted data and returning deferentially-different answers on the outsourced databases. Laplace mechanism are used to illustrate the computation of private queries. Private decision tree learning is also discussed. An experimental evaluation of the proposed solution shows the system incurs small communication and computation overhead. For this reason, the authors in this study [100] proposed a scheme for secure and efficient distribution of big data on BitTorrent networks. The proposed scheme is built inside the BitTorrent protocol and thus allowing the servers to regulate and trace user's behavior and sensitivity of data.

IV. Existing Surveys on Big Data in Cybersecurity

Bertino [4] presented the security and privacy issues for big data concerning the confidentiality, privacy, and trustworthiness. In data confidentiality, the challenges identified were merging large number of access control policies and enforcing control policies in big data sources. Cybersecurity tasks such as user authentication, access control, and user monitoring are noted to be key in identifying threats and stopping them. The author noted that both security and privacy can be achieved by using advanced technologies such as cryptography. Mishra and Singh [5] examined security and privacy challenges associated with big data analytics for protecting database storage and transaction log files, and secure computations in distributed frameworks. The authors in [6] highlighted the benefits of big data analytics and reviewed security and privacy challenges in big data environments using various BDA tools such as Hadoop, MapReduce, and HDFS. Security and privacy challenges associated with big data environments were also listed as random distribution, security of big data computations, and access control. [7] examined big data emerging issues of security and privacy in relation to the use of big data analytic tools such as Hadoop. The work in [8] presented a review of big data security and privacy challenges while storing, searching and analyzing. In [9], the authors conducted a systematic literature review covering security and privacy for big data by categorizing approaches in terms of confidentiality, data integrity, privacy, data analysis, visualization, data format, and stream processing. Miloslavskaya et al. [10] examined the need for Security Operation Centres (SOCs) for organizations that want to achieve the highest protection for their data. The work in [111] looked at security intelligence centres (SICs) for processing of big data. The work in [112] proposed a framework which combined the techniques of security intelligence and big data analytics to support human analysts for prioritization. The work in [113] studied the security issues identified within the field of multimedia applications. In [11], Arora et. al. performed a survey on big data and its security. The work in [114] highlighted the pros of big data, and then later tackles the challenges faced in China. In [115], Zou analyzed major issues associated with big data and especially the breach of personal information, the potential security risks, and the reduction of control rights of users over their personal information.

Mondek et. al. [116] discussed security analytics in this era of big data and the reality of information security. Mahmood and Afzal [12] presented a review of big data analytics trends, tools, and techniques. The study of security analytics is motivated by the inadequacy of existing cybersecurity solutions to counteract cybersecurity attacks associated with big data. Jayasingh, Patra, and Mahesh [117] discussed security issues and challenges that faces security analysts in big data analytics and visualization. In [118], the authors discussed six changes in the information technology sector that they believe will be the game changers for the next 15 years. The work in [13], [14] presented security solutions for the big data in health-care industry. Health-care generates a lot of data from diverse sources and thus making it difficult to analyze. Similarly, in [119], Patil and Seshadri presented security and privacy issues in big data relating to the health-care security policies. The work in [120] summarized the current health-care security scenarios in big data environments in the USA.

The work in [15] put forward a model of big data security service for data providers, users, and cloud service providers. The work in [121] looked at opportunities, challenges, and security concerns associated with the use of big data in cloud computing. Furthermore, the work in [122] proposed integrated auditing for securing big data in the cloud. The authors presented their study by reviewing the characteristics of big data and security challenges in the cloud. The works in [123]–[125] proposed a security measure for big data, virtualization, and the cloud infrastructure and cloud based big data storage systems. Big data is making its way in the power industry. Smart grid has unique characteristics peculiar to it. The work in [126], [127] highlighted different articles that discuss the peculiarities of smart grid big data and how to properly handle it. Authors in [128] looked at security issues brought by big data applications in the telecommunication industry and especially associated with mobile network operators. In [129], authors surveyed three different techniques, namely homomorphic encryption, verifiable computation and multi-party computation. They discuss relevant security threats in the cloud, and a computation model that captures a large class of big data uses cases. The work in [130] studied the impact

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSC.2019.2907247, IEEE Transactions on Services Computing

of security measures on the velocity of the big data system. This research found out that encryption is not an obstacle to the fast and efficient big data processing like it was before because of the introduction of new technologies. They recommended Encryption zones to be set as default in HDFS.

The work in [131] discussed the issues and challenges brought about by the big data deluge; data that is too big, too fast, and too diverse to the extent that are incompatible with the traditional database system. Paryasto et al. [132] presented the security challenges brought about by big data management through NIST risk management framework. The NIST SP800-30 framework provides a guide for conducting risk management on data. The work in [133] discussed the quality assurance for security applications of big data. The interest in quality assurance arises from the lack of confidence in the outcomes of big data applications. The risks in big data analytics arises out of lack of quality assurance. The work in [134] studied an on-line Cauchy based Clustering for cyber attack monitoring. In [135], authors classified big data during it's analysis phase in order to determine the security level of the data currently being analyzed. The work in [136] presented the various kinds of efforts that had taken towards for introducing a context-based information extraction using National Security Information Sources(NSIS) that enlist various kinds of knowledge inspired by natural activities of living things. The work in [137] showed the analysis of 79,012 articles that are published from the year 1916 to 2016 that relates to security and big data privacy.

In [138], the security of personal information on social media in this era of big data was presented. The study looked into the current situation of social network consumer privacy protection and attributed the security problem to personal information leakage and database defects. In [139], authors surveyed pre-processing techniques for data mining using conventional methods such as filtering, imputation, and embedding. The work in [140] discussed the challenges that exist in the era of wireless big data. Finally, the authors in [141] looked at ICT (considered to be the carrier of big data) supply chain security and big data. We provide a comprehensive study of recent research results by categorizing security with big data and big data security. In our work, we explore the role of big data in cybersecurity (as a tool and as an asset). We present up to date literature in this area and we highlight current and foreseeable challenges and trends in this field. We make it easier on readers by summarizing the problem each paper tried to solve and how they approached it in a tabular form.

V. Research Trends and Open Research Challenges

From the first ever virus known as the "creeper" and the first anti-virus made to neutralize it known as the "Reaper", the cybersecurity landscape has changed. The largest insider attack that ever occurred/happened for over a 30 year duration (1976 - 2006) and involved a former Boeing employee stealing intelligent info and handing them to China. Another well known insider threat was the Edward Snowden saga which involved the leakage of classified information from the NSA which resulted in the people distrusting the government. After, another major cyber security attack was Yahoo failing to report that the accounts of over 3 billion users have been jeopardized. Fast forward to 2017, the attack landscape is starting to shift again from data breaches to data being held for ransom. Ransomwares demanding payment (through cryptocurrency) condemn users to the erasure of their data if the ransom is not paid in beginning to gain traction (WannaCry and NotPretya Ransomware). The threat landscape is changing [142] and research trends need to change in other to combat these cybersecurity attacks. Typical trends, open research challenges and problems are shown in Fig. 6 and described below.

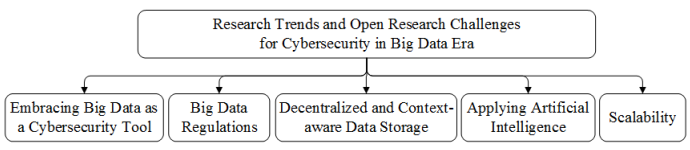

Figure 6. Typical trends, open research challenges and problems.

A. Embracing Big Data as a Cybersecurity Tool

Along with the data generated by IoT devices, the emergence of Bring Your Own Device (BYOD) has made organizations susceptible to various attack vectors. All these devices generate data. Thus organizations are starting to embrace BDA as a tool in their cybersecurity approach. Analyzing the data that passes through the network is essential to protecting the organization. However, some companies still have reservations on employing big data analytics as it tends to be an expensive undertaking. BDA also tends to be a complex field and requires expertise. Furthermore, employees are not comfortable with personal information gathered as this may involve tracking user activity. There are open challenges on how to differentiate the IoT system data, personal data and sensitive data and the protection of each of them using big data analytics.

B. Big Data Regulations

As a result of a myriad data breaches in recent times, new regulations such as Breach of Security Safeguards Regulations in Canada and Europe's General Data Protection Regulation have been implemented. A crucial aspect of the GDPR is the right to be forgotten, which gives an individual the power to enforce the deletion of any information pertaining to him/herself. A research trend we foresee here is self destructing data. Previous work has however been done on this. [143] propose an architecture that aims to solve the issue of personal data privacy. Their research was aimed at protecting the privacy of old data that has been stored on a centralized database which can then be re-used or re-surfaced. Their architecture made sure that copies of such data will become obsolete. This is a research area that might see a lot of growth in years to come, especially due to the emergence of blockchain and decentralized data storage. There are still challenges for big data regulation and policies including the situation where data leaves the organization for cloud storage.

C. Decentralized and Context-aware Data Storage

The most important commodity right now is data. The top companies GAFA (Google, Apple, Facebook, Amazon) have

monopolized data, therefore bringing in the most revenue. New blockchain startups are now basing their business models on how to disrupt these monopolies by highlighting the value of data to the public. The selling point for these startups is that the data stored in a centralized fashion is susceptible to attacks (Facebook, Yahoo, and Equifax hackings) as evident in recent years. A distributed approach to storing data is the safer way to prevent attacks is what is being evangelized. This method is not susceptible to single point of failure problems as well. Companies such as the GAFAs store huge amount of data and they can correctly be termed as data silos. Distributed data storage try to take data away from the hands of these data silos, thereby taking away various security risks. Furthermore, the union of blockchain and big data will make sure that the data that is generated from the blockchain is trustworthy. There are ongoing research and open challenges on decentralized and context-aware data storage for big data.

D. Applying Artificial Intelligence

Artificial Intelligence based polymorphic malware is on the rise. Now, there is an application that can alter malware to trick machine learning antivirus software. In an experiment done by Endgame (a security company), they found out that AI has blind-spots that can be found out by other AI applications. This is evident as seen in Generative Adversarial Networks discovered by google researchers. This shows that organizations should not view machine learning as a fool proof way of defending against malware . More research work is needed in this area because of the rise of GANs. Also, an immediate approach to solve this would be to combine humans and AI in the malware detection approach. AI is not fool proof yet, and we see research trends gearing towards human in the Loop approaches to detect polymorphic malware.

E. Scalability for Cybersecurity Techniques in Big Data era

In big data, protecting everything is hard. The easier approach is to find what is important and protect it. Traditional approaches for securing data might not work in a straightforward way. Thus, finding an optimal approach that is scalable for big data enabled systems is still an active research topic.

VI. Conclusion

In this paper, we have surveyed state of the art literature on big data in cybersecurity. We segmented the work into two parts. The first part was research work involving the use of big data for security purposes. The second part is the research work done on securing big data. We present current trends on the use of BDA as security tool. We also addressed the role of machine learning in this area and some of the challenges machine learning has to overcome before it becomes an important feature in the cybersecurity toolkit. Furthermore, we discussed current literature on techniques used to secure big data. The confidentiality of big data is usually the main focus thus making encryption and access control techniques the main research areas when it comes to big data security. We also discussed the alternative approaches used to secure big data where the proposed approaches rely on other methods than encryption and access control in trying to secure other aspects of the CIA triad. We make it easier on readers by summarizing the problem each paper addresses and their approach to solve it in tabular form. Furthermore, we present future trends in big data security that we foresee, and the challenges associated with it.

References

[1] D. Laney, "3d data management: Controlling data volume, velocity and variety," *META Group Research Note*, vol. 6, no. 70, 2001.

[2] N. Miloslavskaya and A. Tolstoy, "Application of big data, fast data, and data lake concepts to information security issues," in *Future Internet of Things and Cloud Workshops (FiCloudW), IEEE International Conference on*, pp. 148–153, 2016.

[3] D. Rawat and K. Z. Ghafoor, *Smart Cities Cybersecurity and Privacy*. Elsevier, December 2018.

[4] E. Bertino, "Big data-security and privacy," in *Big Data (BigData Congress), 2015 IEEE International Congress on*, pp. 757–761, 2015.

[5] A. D. Mishra and Y. B. Singh, "Big data analytics for security and privacy challenges," in *Computing, Communication and Automation (ICCCA), 2016 International Conference on*, pp. 50–53, 2016.

[6] Y. Gahi, M. Guennoun, and H. T. Mouftah, "Big data analytics: Security and privacy challenges," in *Computers and Communication (ISCC), 2016 IEEE Symposium on*, pp. 952–957, 2016.

[7] K. Abouelmehdi, A. Beni-Hssane, H. Khaloufi, and M. Saadi, "Big data emerging issues: Hadoop security and privacy," in *Multimedia Computing and Systems (ICMCS), 2016 5th International Conference on*, pp. 731–736, 2016.

[8] B. Matturdi, Z. Xianwei, L. Shuai, and L. Fuhong, "Big data security and privacy: A review," *China Communications*, vol. 11, no. 14, pp. 135–145, 2014.

[9] B. Nelson and T. Olovsson, "Security and privacy for big data: A systematic literature review," in *Big Data (Big Data), 2016 IEEE International Conference on*, pp. 3693–3702, 2016.

[10] N. Miloslavskaya, A. Tolstoy, and S. Zapechnikov, "Taxonomy for unsecure big data processing in security operations centers," in *Future Internet of Things and Cloud Workshops (FiCloudW), IEEE International Conference on*, pp. 154–159, 2016.

[11] S. Arora, M. Kumar, P. Johri, and S. Das, "Big heterogeneous data and its security: A survey," in *Computing, Communication and Automation (ICCCA), 2016 International Conference on*, pp. 37–40, 2016.

[12] T. Mahmood and U. Afzal, "Security analytics: Big data analytics for cybersecurity: A review of trends, techniques and tools," in *Information assurance (ncia), 2013 2nd national conference on*, pp. 129–134, 2013.

[13] S. Rao, S. Suma, and M. Sunitha, "Security solutions for big data analytics in healthcare," in *Advances in Computing and Communication Engineering (ICACCE), 2015 Second International Conference on*, pp. 510–514, 2015.

[14] I. Olaronke and O. Oluwaseun, "Big data in healthcare: Prospects, challenges and resolutions," in *Future Technologies Conference (FTC)*, pp. 1152–1157, 2016.

[15] H.-t. Cui, "Research on the model of big data serve security in cloud environment," in *Computer Communication and the Internet (ICCCI), 2016 IEEE International Conference on*, pp. 514–517, 2016.

[16] E. Damiani, "Toward big data risk analysis," in *2015 IEEE International Conference on Big Data (Big Data)*, pp. 1905–1909, 2015.

[17] C. Sinclair, L. Pierce, and S. Matzner, "An application of machine learning to network intrusion detection," in *Computer Security Applications Conference, 1999.(ACSAC'99) Proceedings. 15th Annual*, pp. 371–377, 1999.

[18] E. Chickowski, "A case study in security big data analysis," *Dark Reading*, vol. 9, 2012.

[19] M. C. Raja and M. A. Rabbani, "Big data analytics security issues in data driven information system," *IJIRCCE*, vol. 2, no. 10, 2014.

[20] V. S. Carvalho, M. J. Polidoro, and J. P. Magalhães, "Owlsight: Platform for real-time detection and visualization of cyber threats," in *Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), 2016 IEEE 2nd International Conference on*, pp. 61–66, 2016.

[21] Y. Yao, L. Zhang, J. Yi, Y. Peng, W. Hu, and L. Shi, "A framework for big data security analysis and the semantic technology," in *IT Convergence and Security (ICITCS), 2016 6th International Conference on*, pp. 1–4, 2016.

[22] ProofPoint.com, "The human factor report people-centered threats define the landscape," 2018.

[23] T. Zaki, M. S. Uddin, M. M. Hasan, and M. N. Islam, "Security threats for big data: A study on enron e-mail dataset," in *Research and Innovation in Information Systems (ICRIIS), 2017 International Conference on*, pp. 1–6, 2017.

[24] P. H. Las-Casas, V. S. Dias, W. Meira, and D. Guedes, "A big data architecture for security data and its application to phishing characterization," in *Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), 2016 IEEE 2nd International Conference on*, pp. 36–41, 2016.

[25] A. A. Cárdenas, P. K. Manadhata, and S. Rajan, "Big data analytics for security intelligence," *University of Texas at Dallas@ Cloud Security Alliance*, pp. 1–22, 2013.

[26] W. Jia, "Study on network information security based on big data," in *Measuring Technology and Mechatronics Automation (ICMTMA), 2017 9th International Conference on*, pp. 408–409, 2017.

[27] B. G.-N. Crespo and A. Garwood, "Fighting botnets with cyber-security analytics: Dealing with heterogeneous cyber-security information in new generation siems," in *Availability, Reliability and Security (ARES), 2014 Ninth International Conference on*, pp. 192–198, 2014.

[28] D. C. Le, A. N. Zincir-Heywood, and M. I. Heywood, "Data analytics on network traffic flows for botnet behaviour detection," in *Computational Intelligence (SSCI), 2016 IEEE Symposium Series on*, pp. 1–7, 2016.

[29] H. Fatima, S. Satpathy, S. Mahapatra, G. Dash, and S. K. Pradhan, "Data fusion & visualization application for network forensic investigation-a case study," in *Anti-Cyber Crimes (ICACC), 2017 2nd International Conference on*, pp. 252–256, 2017.

[30] K. Gai, M. Qiu, and S. A. Elnagdy, "A novel secure big data cyber incident analytics framework for cloud-based cybersecurity insurance," in *Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), 2016 IEEE 2nd International Conference on*, pp. 171–176, 2016.

[31] K. Gai, M. Qiu, and S. A. Elnagdy, "Security-aware information classifications using supervised learning for cloud-based cyber risk management in financial big data," in *Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), 2016 IEEE 2nd International Conference on*, pp. 197–202, 2016.

[32] G. Gardikis, K. Tzoulas, K. Tripolitis, A. Bartzas, S. Costicoglou, A. Lioy, B. Gaston, C. Fernandez, C. Davila, A. Litke, *et al.*, "Shield: A novel nfv-based cybersecurity framework," in *Network Softwarization (NetSoft), 2017 IEEE Conference on*, pp. 1–6, 2017.

[33] I. Saenko, I. Kotenko, and A. Kushnerevich, "Parallel processing of big heterogeneous data for security monitoring of iot networks," in *Parallel, Distributed and Network-based Processing (PDP), 2017 25th Euromicro International Conference on*, pp. 329–336, 2017.

[34] F. Gottwalt and A. P. Karduck, "Sim in light of big data," in *Innovations in Information Technology (IIT), 2015 11th International Conference on*, pp. 326–331, 2015.

[35] T. Y. Win, H. Tianfield, and Q. Mair, "Big data based security analytics for protecting virtualized infrastructures in cloud computing," *IEEE Transactions on Big Data*, 2017.

[36] C. Puri and C. Dukatz, "Analyzing and predicting security event anomalies: Lessons learned from a large enterprise big data streaming analytics deployment," in *Database and Expert Systems Applications (DEXA), 2015 26th International Workshop on*, pp. 152–158, 2015.

[37] C. Liu, Y. Cai, and T. Wang, "Security evaluation of rc4 using big data analytics," in *Software Engineering and Service Science (ICSESS), 2016 7th IEEE International Conference on*, pp. 316–320, 2016.

[38] K. Scarfone and P. Mell, "Guide to intrusion detection and prevention systems (idps)," *NIST special publication*, vol. 800, no. 2007, p. 94, 2007.

[39] S. Mukkamala, A. Sung, and A. Abraham, "Cyber security challenges: Designing efficient intrusion detection systems and antivirus tools," *Vemuri, V. Rao, Enhancing Computer Security with Smart Technology.(Auerbach, 2006)*, pp. 125–163, 2005.

[40] S. Suthaharan, "Big data classification: Problems and challenges in network intrusion prediction with machine learning," *ACM SIGMETRICS Performance Evaluation Review*, vol. 41, no. 4, pp. 70–73, 2014.

[41] H.-M. Chen, R. Kazman, I. Monarch, and P. Wang, "Predicting and fixing vulnerabilities before they occur: a big data approach," in *Proceedings of the 2nd ACM International Workshop on BIG Data Software Engineering*, pp. 72–75, 2016.

[42] S. Marchal, X. Jiang, R. State, and T. Engel, "A big data architecture for large scale security monitoring," in *Big data (BigData Congress), 2014 IEEE international congress on*, pp. 56–63, 2014.

[43] B. Yang and T. Zhang, "A scalable meta-model for big data security analyses," in *Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), 2016 IEEE 2nd International Conference on*, pp. 55–60, 2016.

[44] P. Casas, F. Soro, J. Vanerio, G. Settanni, and A. D'Alconzo, "Network security and anomaly detection with big-dama, a big data analytics framework," in *Cloud Networking (CloudNet), 2017 IEEE 6th International Conference on*, pp. 1–7, 2017.

[45] S. Kumar, A. Viinikainen, and T. Hamalainen, "Machine learning classification model for network based intrusion detection system," in *Internet Technology and Secured Transactions (ICITST), 2016 11th International Conference for*, pp. 242–249, 2016.

[46] U. Fiore, F. Palmieri, A. Castiglione, and A. De Santis, "Network anomaly detection with the restricted boltzmann machine," *Elsevier Neurocomputing*, vol. 122, pp. 13–23, 2013.

[47] M. A. Salama, H. F. Eid, R. A. Ramadan, A. Darwish, and A. E. Hassanien, "Hybrid intelligent intrusion detection scheme," in *Soft computing in industrial applications*, pp. 293–303, Springer, 2011.

[48] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system," in *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS)*, pp. 21–26, ICST (Institute for Computer Sciences, Social-Informatics and ..., 2016.

[49] M. M. Najafabadi, F. Villanustre, T. M. Khoshgoftaar, N. Seliya, R. Wald, and E. Muharemagic, "Deep learning applications and challenges in big data analytics," *Journal of Big Data*, vol. 2, no. 1, p. 1, 2015.

[50] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013.

[51] N. Papernot, P. McDaniel, A. Sinha, and M. Wellman, "Towards the science of security and privacy in machine learning," *arXiv preprint arXiv:1611.03814*, 2016.

[52] W. Xu, D. Evans, and Y. Qi, "Feature squeezing: Detecting adversarial examples in deep neural networks," *arXiv preprint arXiv:1704.01155*, 2017.

[53] V. Patel, "A practical solution to improve cyber security on a global scale," in *Cybersecurity Summit (WCS), 2012 Third Worldwide*, pp. 1–5, 2012.

[54] C. Zhong, J. Yen, P. Liu, and R. F. Erbacher, "Automate cybersecurity data triage by leveraging human analysts' cognitive process," in *Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), 2016 IEEE 2nd International Conference on*, pp. 357–363, 2016.

[55] M. G. Schultz, E. Eskin, F. Zadok, and S. J. Stolfo, "Data mining methods for detection of new malicious executables," in *Security and Privacy, 2001. S&P 2001. Proceedings. 2001 IEEE Symposium on*, pp. 38–49, 2001.

[56] H. H. Huang and H. Liu, "Big data machine learning and graph analytics: Current state and future challenges," in *Big Data (Big Data), 2014 IEEE International Conference on*, pp. 16–17, 2014.

[57] N. Naik, P. Jenkins, N. Savage, and V. Katos, "Big data security analysis approach using computational intelligence techniques in r for desktop users," in *Computational Intelligence (SSCI), 2016 IEEE Symposium Series on*, pp. 1–8, 2016.

[58] K. Kaur, A. Syed, A. Mohammad, and M. N. Halgamuge, "An evaluation of major threats in cloud computing associated with big data," in *Big Data Analysis (ICBDA), 2017 IEEE 2nd International Conference on*, pp. 368–372, 2017.

[59] J. Kepner, V. Gadepally, P. Michaleas, N. Schear, M. Varia, A. Yerukhimovich, and R. K. Cunningham, "Computing on masked data: a high performance method for improving big data veracity," in *High Performance Extreme Computing Conference (HPEC), 2014 IEEE*, pp. 1–6, 2014.

[60] D. Wang, B. Guo, Y. Shen, S.-J. Cheng, and Y.-H. Lin, "A faster fully homomorphic encryption scheme in big data," in *Big Data Analysis (ICBDA), 2017 IEEE 2nd International Conference on*, pp. 345–349, 2017.

[61] T. B. Patil, G. K. Patnaik, and A. T. Bhole, "Big data privacy using fully homomorphic non-deterministic encryption," in *Advance Computing Conference (IACC), 2017 IEEE 7th International*, pp. 138–143, 2017.

[62] B. Cui, B. Zhang, and K. Wang, "A data masking scheme for sensitive big data based on format-preserving encryption," in *Computational Science and Engineering (CSE) and Embedded and Ubiquitous Computing (EUC), 2017 IEEE International Conference on*, vol. 1, pp. 518–524, 2017.

[63] M. Li, S. Yu, K. Ren, and W. Lou, "Securing personal health records in cloud computing: Patient-centric and fine-grained data access control in multi-owner settings.," in *SecureComm*, vol. 10, pp. 89–106, Springer, 2010.

[64] T. Yang, P. Shen, X. Tian, and C. Chen, "A fine-grained access control scheme for big data based on classification attributes," in *Distributed Computing Systems Workshops (ICDCSW), 2017 IEEE 37th International Conference on*, pp. 238–245, 2017.

[65] S. Pérez, J. L. Hernández-Ramos, D. Pedone, D. Rotondi, L. Straniero, and A. F. Skarmeta, "A digital envelope approach using attribute-based encryption for secure data exchange in iot scenarios," in *Global Internet of Things Summit (GIoTS), 2017*, pp. 1–6, 2017.

[66] G. Xu, Y. Ren, H. Li, D. Liu, Y. Dai, and K. Yang, "Cryptmdb: A practical encrypted mongodb over big data," in *Communications (ICC), 2017 IEEE International Conference on*, pp. 1–6, 2017.

[67] A. Al Mamun, K. Salah, S. Al-maadeed, and T. R. Sheltami, "Bigcrypt for big data encryption," in *Software Defined Systems (SDS), 2017 Fourth International Conference on*, pp. 93–99, 2017.

[68] X. Dong, R. Li, H. He, W. Zhou, Z. Xue, and H. Wu, "Secure sensitive data sharing on a big data platform," *Tsinghua Science and Technology*, vol. 20, no. 1, pp. 72–80, 2015.

[69] A. Sharma and D. Sharma, "Big data protection via neural and quantum cryptography," in *Computing for Sustainable Global Development (INDIACom), 2016 3rd International Conference on*, pp. 3701–3704, 2016.

[70] C. Zhao and J. Liu, "Novel group key transfer protocol for big data security," in *Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), 2015 IEEE*, pp. 161–165, 2015.

[71] S. Almuhammadi and A. Amro, "Double-hashing operation mode for encryption," in *Computing and Communication Workshop and Conference (CCWC), 2017 IEEE 7th Annual*, pp. 1–7, 2017.

[72] F. A. Kadhim, G. H. Abdul-Majeed, and R. S. Ali, "Enhancement cast block algorithm to encrypt big data," in *New Trends in Information & Communications Technology Applications (NTICT), 2017 Annual Conference on*, pp. 80–85, 2017.

[73] A. Kulkarni, C. Shea, H. Homayoun, and T. Mohsenin, "Less: Big data sketching and encryption on low power platform," in *Proceedings of the Conference on Design, Automation & Test in Europe*, pp. 1635–1638, European Design and Automation Association, 2017.

[74] C. Xiao, L. Wang, Z. Jie, and T. Chen, "A multi-level intelligent selective encryption control model for multimedia big data security in sensing system with resource constraints," in *Cyber Security and Cloud Computing (CSCloud), 2016 IEEE 3rd International Conference on*, pp. 148–153, 2016.

[75] A. Soceanu, M. Vasylenko, A. Egner, and T. Muntean, "Managing the privacy and security of ehealth data," in *Control Systems and Computer Science (CSCS), 2015 20th International Conference on*, pp. 439–446, 2015.

[76] A. Al-Shomrani, F. Fathy, and K. Jambi, "Policy enforcement for big data security," in *Anti-Cyber Crimes (ICACC), 2017 2nd International Conference on*, pp. 70–74, 2017.

[77] A. Samuel, M. I. Sarfraz, H. Haseeb, S. Basalamah, and A. Ghafoor, "A framework for composition and enforcement of privacy-aware and context-driven authorization mechanism for multimedia big data," *IEEE Transactions on Multimedia*, vol. 17, no. 9, pp. 1484–1494, 2015.

[78] F. Gao, "Research on cloud security control mechanism based on big data," in *Smart Grid and Electrical Automation (ICSGEA), 2017 International Conference on*, pp. 366–370, 2017.

[79] A. Gupta, A. Verma, P. Kalra, and L. Kumar, "Big data: A security compliance model," in *IT in Business, Industry and Government (CSIBIG), 2014 Conference on*, pp. 1–5, 2014.

[80] N.-Y. Lee and B.-H. Wu, "Privacy protection technology and access control mechanism for medical big data," in *2017 6th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)*, pp. 424–429, 2017.

[81] M. R. Islam, M. Habiba, and M. I. I. Kashem, "A framework for providing security to personal healthcare records," in *Networking, Systems and Security (NSysS), 2017 International Conference on*, pp. 168–173, 2017.

[82] R. Achana, R. S. Hegadi, and T. Manjunath, "A novel data security framework using e-mod for big data," in *Electrical and Computer Engineering (WIECON-ECE), 2015 IEEE International WIE Conference on*, pp. 546–551, 2015.

[83] S.-H. Kim, N.-U. Kim, and T.-M. Chung, "Attribute relationship evaluation methodology for big data security," in *IT Convergence and Security (ICITCS), 2013 International Conference on*, pp. 1–4, 2013.

[84] S.-H. Kim, J.-H. Eom, and T.-M. Chung, "Big data security hardening methodology using attributes relationship," in *Information Science and Applications (ICISA), 2013 International Conference on*, pp. 1–2, 2013.

[85] H. A. Al Hamid, S. M. M. Rahman, M. S. Hossain, A. Almogren, and A. Alamri, "A security model for preserving the privacy of medical big data in a healthcare cloud using a fog computing facility with pairing-based cryptography," *IEEE Access*, vol. 5, pp. 22313–22328, 2017.

[86] A. Unal, "Security in big data of medical records," in *IT in Business, Industry and Government (CSIBIG), 2014 Conference on*, pp. 1–2, 2014.

[87] C. Ye, Z. Xiong, Y. Ding, J. Li, G. Wang, X. Zhang, and K. Zhang, "Secure multimedia big data sharing in social networks using fingerprinting and encryption in the jpeg2000 compressed domain," in *Trust, Security and Privacy in Computing and Communications (TrustCom), 2014 IEEE 13th International Conference on*, pp. 616–621, 2014.

[88] J. Wu, K. Ota, M. Dong, J. Li, and H. Wang, "Big data analysis based security situational awareness for smart grid," *IEEE Transactions on Big Data*, 2016.

[89] S. Pissanetzky, "On the future of information: Reunification, computability, adaptation, cybersecurity, semantics," *IEEE Access*, vol. 4, pp. 1117–1140, 2016.

[90] L. Xu, P. D. Khoa, S. H. Kim, W. W. Ro, and W. Shi, "Another look at secure big data processing: Formal framework and a potential approach," in *Cloud Computing (CLOUD), 2015 IEEE 8th International Conference on*, pp. 548–555, 2015.

[91] K. Gai, M. Qiu, and H. Zhao, "Security-aware efficient mass distributed storage approach for cloud systems in big data," in *Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), 2016 IEEE 2nd International Conference on*, pp. 140–145, 2016.

[92] L. Xu, H. Kim, X. Wang, W. Shi, and T. Suh, "Privacy preserving large scale dna read-mapping in mapreduce framework using fpgas," in *Field Programmable Logic and Applications (FPL), 2014 24th International Conference on*, pp. 1–4, 2014.

[93] G. Collard, E. Disson, G. Talens, and S. Ducroquet, "Proposition of a method to aid security classification in cybersecurity context," in *Privacy, Security and Trust (PST), 2016 14th Annual Conference on*, pp. 88–95, 2016.

[94] P. Adluru, S. S. Datla, and X. Zhang, "Hadoop eco system for big data security and privacy," in *Systems, Applications and Technology Conference (LISAT), 2015 IEEE Long Island*, pp. 1–6, 2015.

[95] J. Singh, "Real time big data analytic: Security concern and challenges with machine learning algorithm," in *IT in Business, Industry and Government (CSIBIG), 2014 Conference on*, pp. 1–4, 2014.

[96] T. Yang and S. Jia, "Research on network security visualization under big data environment," in *Computer Symposium (ICS), 2016 International*, pp. 660–662, 2016.

[97] T. Ban, S. Pang, M. Eto, D. Inoue, K. Nakao, and R. Huang, "Towards early detection of novel attack patterns through the lens of a large-scale darknet," in *Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld), 2016 Intl IEEE Conferences*, pp. 341–349, 2016.

[98] S. A. Alsuhibany, "A space-and-time efficient technique for big data security analytics," in *Information Technology (Big Data Analysis)(KACSTIT), Saudi International Conference on*, pp. 1–6, 2016.

[99] N. Mohammed, S. Barouti, D. Alhadidi, and R. Chen, "Secure and private management of healthcare databases for data mining," in *Computer-Based Medical Systems (CBMS), 2015 IEEE 28th International Symposium on*, pp. 191–196, 2015.

[100] L. Xiao, C. Xu, J. Qin, G. Qin, M. Zhu, L. Ruan, Z. Wang, M. Li, and D. Tan, "Secure distribution of big data based on bittorrent," in *Dependable, Autonomic and Secure Computing (DASC), 2013 IEEE 11th International Conference on*, pp. 82–90, 2013.

[101] W. Zhijun and W. Caiyun, "Security-as-a-service in big data of civil aviation," in *Computer and Communications (ICCC), 2015 IEEE International Conference on*, pp. 240–244, 2015.

[102] C. Hu and Y. Huo, "Efficient privacy-preserving dot-product computation for mobile big data," *IET Communications*, vol. 11, no. 5, pp. 704–712, 2016.

[103] S. Q. Ren, T. H. Meng, N. Yibin, and K. M. M. Aung, "Privacy-preserved multi-party data merging with secure equality evaluation," in *Cloud Computing Research and Innovations (ICCCRI), 2016 International Conference on*, pp. 34–41, 2016.

[104] K. Benzidane, H. El Alloussi, O. El Warrak, L. Fetjah, S. J. Andaloussi, and A. Sekkaki, "Toward a cloud-based security intelligence with big data processing," in *Network Operations and Management Symposium (NOMS), 2016 IEEE/IFIP*, pp. 1089–1092, 2016.

[105] Z.-W. Lu, "Research about new media security technology base on big data era," in *Dependable, Autonomic and Secure Computing, 14th Intl Conf on Pervasive Intelligence and Computing, 2nd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), 2016 IEEE 14th Intl C*, pp. 933–936, 2016.

[106] L. Yue, H. Junqin, Q. Shengzhi, and W. Ruijin, "Big data model of security sharing based on blockchain," in *Big Data Computing and Communications (BIGCOM), 2017 3rd International Conference on*, pp. 117–121, 2017.

[107] M. Tang, M. Alazab, and Y. Luo, "Big data for cybersecurity: Vulnerability disclosure trends and dependencies," *IEEE Transactions on Big Data*, 2017.

[108] P. A. Dhande and A. Kadam, "New approach for load rebalancer, scheduler in big data with security mechanism in cloud environment," in *Advances in Electronics, Communication and Computer Technology (ICAECCT), 2016 IEEE International Conference on*, pp. 247–250, 2016.

[109] D. Puthal, N. Malik, S. P. Mohanty, E. Kougianos, and C. Yang, "The blockchain as a decentralized security framework," *IEEE Consum. Electron. Mag.*, vol. 7, no. 2, pp. 18–21, 2018.

[110] Q. Xia, E. B. Sifah, K. O. Asamoah, J. Gao, X. Du, and M. Guizani, "Medshare: Trust-less medical data sharing among cloud service providers via blockchain," *IEEE Access*, vol. 5, pp. 14757–14767, 2017.

[111] N. Miloslavskaya, "Security intelligence centers for big data processing," in *Future Internet of Things and Cloud Workshops (FiCloudW), 2017 5th International Conference on*, pp. 7–13, 2017.

[112] M. Marchetti, F. Pierazzi, A. Guido, and M. Colajanni, "Countering advanced persistent threats through security intelligence and big data analytics," in *Cyber Conflict (CyCon), 2016 8th International Conference on*, pp. 243–261, 2016.

[113] Q. Jin, Y. Xiang, G. Sun, Y. Liu, and C.-C. Chang, "Cybersecurity for cyber-enabled multimedia applications," *IEEE MultiMedia*, vol. 24, no. 4, pp. 10–13, 2017.

[114] Y. Mengke, Z. Xiaoguang, Z. Jianqiu, and X. Jianjian, "Challenges and solutions of information security issues in the age of big data," *China Communications*, vol. 13, no. 3, pp. 193–202, 2016.

[115] H. Zou, "Protection of personal information security in the age of big data," in *Computational Intelligence and Security (CIS), 2016 12th International Conference on*, pp. 586–589, 2016.

[116] D. Mondek, R. B. Blažek, and T. Zahradnickỳ, "Security analytics in the big data era," in *Software Quality, Reliability and Security Companion (QRS-C), 2017 IEEE International Conference on*, pp. 605–606, 2017.

[117] B. B. Jayasingh, M. Patra, and D. B. Mahesh, "Security issues and challenges of big data analytics and visualization," in *Contemporary Computing and Informatics (IC3I), 2016 2nd International Conference on*, pp. 204–208, 2016.

[118] A. Kott, A. Swami, and P. McDaniel, "Security outlook: six cyber game changers for the next 15 years," *Computer*, vol. 47, no. 12, pp. 104–106, 2014.

[119] H. K. Patil and R. Seshadri, "Big data security and privacy issues in healthcare," in *Big Data (BigData Congress), 2014 IEEE International Congress on*, pp. 762–765, 2014.

[120] S. Chandra, S. Ray, and R. Goswami, "Big data security in healthcare: Survey on frameworks and algorithms," in *Advance Computing Conference (IACC), 2017 IEEE 7th International*, pp. 89–94, 2017.

[121] S. Anandaraj and M. Kemal, "Research opportunities and challenges of security concerns associated with big data in cloud computing," in *I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), 2017 International Conference on*, pp. 746–751, 2017.

[122] Y. Wang, B. Rawal, and Q. Duan, "Securing big data in the cloud with integrated auditing," in *Smart Cloud (SmartCloud), 2017 IEEE International Conference on*, pp. 126–131, 2017.

[123] S. Bahulikar, "Security measures for the big data, virtualization and the cloud infrastructure," in *Information Processing (IICIP), 2016 1st India International Conference on*, pp. 1–4, 2016.

[124] A. Sharif, S. Cooney, S. Gong, and D. Vitek, "Current security threats and prevention measures relating to cloud services, hadoop concurrent processing, and big data," in *Big Data (Big Data), 2015 IEEE International Conference on*, pp. 1865–1870, IEEE, 2015.

[125] Z. Tan, U. T. Nagar, X. He, P. Nanda, R. P. Liu, S. Wang, and J. Hu, "Enhancing big data security with collaborative intrusion detection," *IEEE cloud computing*, vol. 1, no. 3, pp. 27–33, 2014.

[126] J. Zhao, Y. Wang, and Y. Xia, "Analysis of information security of electric power big data and its countermeasures," in *Computational Intelligence and Security (CIS), 2016 12th International Conference on*, pp. 243–248, 2016.

[127] J. Hu and A. V. Vasilakos, "Energy big data analytics and security: challenges and opportunities," *IEEE Transactions on Smart Grid*, vol. 7, no. 5, pp. 2423–2436, 2016.

[128] C. Dincer, G. Akpolat, and E. Zeydan, "Security issues of big data applications served by mobile operators," in *Signal Processing and Communications Applications Conference (SIU), 2017 25th*, pp. 1–4, 2017.

[129] S. Yakoubov, V. Gadepally, N. Schear, E. Shen, and A. Yerukhimovich, "A survey of cryptographic approaches to securing big-data analytics in the cloud," in *High Performance Extreme Computing Conference (HPEC), 2014 IEEE*, pp. 1–6, 2014.

[130] L. Dupré and Y. Demchenko, "Impact of information security measures on the velocity of big data infrastructures," in *High Performance Computing & Simulation (HPCS), 2016 International Conference on*, pp. 492–500, 2016.

[131] N. Chaudhari and S. Srivastava, "Big data security issues and challenges," in *Computing, Communication and Automation (ICCCA), 2016 International Conference on*, pp. 60–64, 2016.

[132] M. Paryasto, A. Alamsyah, B. Rahardjo, *et al.*, "Big-data security management issues," in *Information and Communication Technology (ICoICT), 2014 2nd International Conference on*, pp. 59–63, 2014.

[133] R. Clarke, "Quality assurance for security applications of big data," in *Intelligence and Security Informatics Conference (EISIC), 2016 European*, pp. 1–8, 2016.

[134] I. Škrjanc, A. S. de Miguel, J. A. Iglesias, A. Ledezma, and D. Dovžan, "Evolving cauchy possibilistic clustering based on cosine similarity for monitoring cyber systems," in *Evolving and Adaptive Intelligent Systems (EAIS), 2017*, pp. 1–5, 2017.

[135] S. Alouneh, I. Hababeh, F. Al-Hawari, and T. Alajrami, "Innovative methodology for elevating big data analysis and security," in *Open Source Software Computing (OSSCOM), 2016 2nd International Conference on*, pp. 1–5, 2016.

[136] K. Dhanasekaran and B. Surendiran, "Nature-inspired classification for mining social space information: National security intelligence and big data perspective," in *Green Engineering and Technologies (IC-GET), 2016 Online International Conference on*, pp. 1–6, 2016.

[137] K. D. Strang and Z. Sun, "Meta-analysis of big data security and privacy: Scholarly literature gaps," in *Big Data (Big Data), 2016 IEEE International Conference on*, pp. 4035–4037, 2016.

[138] L. Yuqing, "Research on personal information security on social network in big data era," in *Smart Grid and Electrical Automation (ICSGEA), 2017 International Conference on*, pp. 676–678, 2017.

[139] J. Hariharakrishnan, S. Mohanavalli, K. S. Kumar, *et al.*, "Survey of pre-processing techniques for mining big data," in *Computer, Communication and Signal Processing (ICCCSP), 2017 International Conference on*, pp. 1–5, 2017.

[140] S. Bi, R. Zhang, Z. Ding, and S. Cui, "Wireless communications in the era of big data," *IEEE communications magazine*, vol. 53, no. 10, pp. 190–199, 2015.

[141] T. Lu, X. Guo, B. Xu, L. Zhao, Y. Peng, and H. Yang, "Next big thing in big data: the security of the ict supply chain," in *Social Computing (SocialCom), 2013 International Conference on*, pp. 1066–1073, 2013.

[142] E. Damiani, C. Ardagna, F. Zavatarelli, E. Rekleitis, and L. Marinos, "Big Data Threat Landscape," *European Union Agency for Network and Information Security*, Jan 2017. web: https://www.enisa.europa.eu/publications/bigdata-threat-landscape.

[143] R. Geambasu, T. Kohno, A. A. Levy, and H. M. Levy, "Vanish: Increasing data privacy with self-destructing data.," in *USENIX Security Symposium*, vol. 316, 2009.