# Design and Evaluation of a Forensic-Ready Framework for Smart Classrooms

Henry Rossi Andrian[1], Suhardi[2], I Gusti Bagus Baskara Nugraha[3]
School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Bandung, Indonesia[1, 2, 3]

***Abstract*—The rise of cyber threats in educational environments underscores the need for forensic-ready systems tailored to digital learning platforms like smart classrooms. This study proposes a proactive forensic-ready framework that integrates threat estimation, risk profiling, data identification, and collection management into a continuous readiness cycle. Blockchain technology ensures log immutability, while LMS APIs enable systematic evidence capture with minimal disruption to learning processes. Monte Carlo Simulation validates the framework's performance across key metrics. Results show a log capture success rate of 77.27%, with high accuracy for structured attacks such as SQL Injection. The system maintains operational efficiency, adding only 15% average CPU overhead. Forensic logs are securely stored in JSON format on a blockchain ledger, ensuring both integrity and accessibility. However, reduced effectiveness for complex attacks like Remote Code Execution and occasional retrieval delays under heavy loads highlight areas for improvement. Future enhancements will focus on expanding threat coverage and optimizing log retrieval. By addressing vulnerabilities unique to smart classrooms, such as unauthorized access and data manipulation, this study introduces a scalable, domain-specific solution for enhancing forensic readiness and cybersecurity in educational ecosystems.**

***Keywords*—*Forensic-ready system; smart classroom; threat estimation; risk profile***

## I. INTRODUCTION

Digital forensics has become a cornerstone of modern cybersecurity and law enforcement, addressing the urgent need to locate, preserve, and analyze digital evidence in response to cybercrimes, data breaches, and security violations. With cyberattack-related damages reaching an estimated $10.3 billion in 2022, according to the FBI, the demand for robust and adaptive forensic systems is more pressing than ever. Over time, the field has evolved significantly, giving rise to various definitions and frameworks. The National Institute of Standards and Technology (NIST) defines digital forensics as the application of scientific and engineering methods to collect, preserve, analyze, and present digital evidence. Likewise, scholars such as [1] and [2] highlight data recovery, analysis, and preservation as core pillars of effective forensic practice. However, despite ongoing advancements, most current methodologies remain reactive—emphasizing post-incident investigation rather than proactive evidence acquisition—which limits their effectiveness against the scale and speed of today's cyber threats.

A major challenge confronting digital forensics today is the need to adapt to evolving data storage and processing technologies. Traditional evidence sources such as hard drives and RAM are increasingly being replaced by cloud-native infrastructures and decentralized systems like blockchain. While these technologies offer enhanced scalability and efficiency, they introduce complex forensic barriers. For example, cloud environments often lack a clear physical boundary, complicating evidence acquisition and chain of custody, whereas blockchain systems distribute data across global nodes, creating technical and jurisdictional hurdles. These emerging complexities underscore the inadequacy of traditional forensic approaches and call for new frameworks that are intrinsically designed to operate within modern digital ecosystems.

In response to these technological disruptions, the concept of forensic-ready systems has gained attention as a proactive approach to digital evidence management. Unlike traditional post-incident forensic methods, forensic-ready systems aim to ensure that critical digital evidence is systematically captured, preserved, and made readily available without interrupting operational workflows. These systems are designed to integrate forensic capabilities directly into live environments, balancing investigative needs with system performance. However, current research in this area—such as that by [3] and [4]—often focuses on high-level architectural models or broad organizational policies, lacking domain-specific implementations that address the unique operational and technical requirements of contexts like digital education platforms.

To address this gap, this study introduces a novel forensic-ready system framework specifically tailored for smart classrooms—an environment increasingly dependent on interconnected digital learning platforms and therefore highly susceptible to cyber threats. The proposed framework incorporates threat assessment, risk profiling, and proactive data requirement analysis to ensure forensic evidence is continuously and efficiently captured. It also integrates blockchain technology to maintain data integrity, ensure transparency, and secure digital logs in a tamper-resistant format. By embedding forensic readiness into the operational fabric of smart classrooms, this framework not only supports proactive incident response but also minimizes disruption to the learning process, thereby addressing both the technical challenges outlined earlier and the limitations of current generalized approaches.

Ultimately, this study advances both theoretical understanding and practical implementation of forensic-ready systems. By applying the proposed framework within smart classroom environments, it enhances cybersecurity resilience in digital education while offering a scalable model adaptable to other digital ecosystems. This work represents a critical step towards aligning forensic methodologies with contemporary

technological landscapes, contributing valuable insights and innovative solutions to the field of digital forensics.

The remainder of the study is structured as follows: Section II reviews relevant literature, highlighting critical gaps in existing forensic-ready methodologies and their applications in educational contexts. Section III describes the proposed forensic-ready framework and its implementation methodology, including the integration of blockchain and proactive forensic mechanisms. Section IV presents simulation results using Monte Carlo Simulation (MCS), followed by detailed discussions comparing these findings with related studies. Section V concludes the study, summarizing key insights and suggesting directions for future research.

## II. Related Work

Digital forensics involves systematic collection, preservation, and analysis of digital evidence to ensure its integrity and legal admissibility. As a critical component of modern cybersecurity, digital forensics enables organizations to investigate security incidents, trace the origins of attacks, and support legal proceedings. Recent advancements have introduced artificial intelligence (AI) and machine learning algorithms, enabling automated anomaly detection, efficient analysis of large datasets, and predictive threat modeling[5]. Additionally, the proliferation of cloud computing and Internet of Things (IoT) devices has spurred the development of cloud forensic readiness frameworks and decentralized evidence storage techniques, reducing the time and cost associated with traditional investigations while improving scalability and responsiveness in distributed environments [6] [7].

Proactive forensics extends these capabilities by embedding forensic functions into the operational fabric of systems, allowing evidence to be collected preemptively—before a security breach escalates. This approach reduces the burden of post-incident investigations and improves overall cyber resilience [8][6]. Technological innovations are key to this paradigm shift. AI-powered behavioral analysis and anomaly detection help identify suspicious activity in real time[5], while blockchain technology ensures the immutability and traceability of forensic data[9]. Dynamic logging systems, particularly in cloud and IoT infrastructures, further enhance proactive forensics by enabling real-time evidence collection across distributed nodes. These developments are redefining proactive forensics as a foundational element in cybersecurity architecture.

Closely related is the concept of forensic readiness, which emphasizes an organization's ability to efficiently capture and preserve digital evidence with minimal disruption. Emerging forensic readiness models incorporate cloud-native architectures, centralized logging mechanisms, and edge computing to enhance real-time data collection at the point of origin [10][11][12]. Blockchain ensures the authenticity and permanence of logs [9], while AI automates artifact classification and prioritization, leading to faster and more accurate investigations. Together, these technologies create robust environments, where digital evidence is securely maintained and readily available, thereby improving both the quality of investigations and compliance with regulatory standards.

Building upon these innovations, forensic-ready systems have evolved to integrate sophisticated monitoring tools and advanced evidence-preservation protocols. These systems leverage machine learning for automated detection of high-value forensic events and utilize blockchain to maintain tamper-proof logs [3][13][14][8][6]. Edge computing capabilities also enable decentralized logging and analysis, which proves particularly beneficial in latency-sensitive and distributed environments, such as IoT ecosystems[15]. As a result, forensic-ready systems are becoming highly adaptive solutions that can meet the complex demands of modern cybersecurity environments.

Although significant advancements have been made in digital forensics, applying these technologies to smart classrooms using Learning Management Systems (LMS) presents unique challenges. The rapid digitalization of education has increased exposure to threats such as unauthorized access, data breaches, and malware attacks. LMS platforms must now incorporate robust cybersecurity mechanisms to protect sensitive student data and ensure system reliability. Issues like denial-of-service attacks, weak user authentication, and data manipulation have prompted the adoption of enhanced safeguards such as multi-factor authentication, dynamic access control, and real-time monitoring [16][17][18]. These security measures highlight the growing need for specialized frameworks that address the vulnerabilities inherent in LMS environments.

To complement preventive security measures, digital forensics plays a crucial role in LMS-based smart classrooms by enabling the investigation of incidents that bypass security defenses. In this context, forensic processes include automated collection, preservation, and analysis of evidence related to user activity, system behavior, and potential breaches. Recent advances in forensic readiness have enabled real-time logging and incident tracking within LMS platforms, improving the traceability and integrity of digital evidence [16] [18][19]. These frameworks support rapid investigation and response, ensuring that educational institutions can identify vulnerabilities, enforce accountability, and enhance the overall resilience of smart learning environments.

Effectively addressing the identified challenges necessitates a comprehensive cybersecurity strategy that integrates digital, proactive, and forensic-ready components specifically adapted to LMS-based smart classrooms. These integrated systems utilize real-time logging, AI-enabled anomaly detection[5], blockchain-secured data integrity[9], and centralized forensic dashboards to safeguard educational infrastructures. In addition to enhancing protection, such systems streamline forensic workflows and minimize response times, thereby reinforcing forensic readiness as a foundational element of secure and resilient educational technology environments.

Monte Carlo Simulation (MCS) is a popular method for validating integrated forensic-ready systems. MCS simulates multiple cybersecurity threat scenarios using probabilistic modeling and random sampling to assess system performance in response time, log retrieval accuracy, and evidence integrity. MCS provides statistically meaningful insights into system behavior under uncertainty by repeating iterations, making it useful for testing forensic-ready frameworks before deployment. In digital forensics, [20] used MCS with the Analytic Hierarchy

Process (AHP) to support risk analysis in security management systems, and [21] used Monte Carlo Feature Selection to validate network-based forensic artifacts. These examples demonstrate MCS's capacity to verify forensic system dependability and preparation in complex, dynamic environments like LMS-based smart classrooms.

Despite recent advancements, existing forensic-ready frameworks show key limitations when applied to smart classrooms. Many focus on high-level policies or enterprise systems and lack the domain-specific features needed for educational platforms with dynamic user interactions, multiple access levels, and real-time activity [18][22]. Most also lack real-time evidence capture, which is vital for responding to time-sensitive academic incidents. Furthermore, the limited adoption of immutable logging mechanisms such as blockchain weakens log integrity and reduces the evidentiary value of collected data [9]. Recognizing these shortcomings, recent studies have explored the benefits of tailoring digital forensic readiness (DFR) frameworks to specific operational domains, such as Industrial IoT [23], e-Government [24], wireless medical systems [25], and software-defined networks [26], demonstrating that alignment with domain-specific architectures, workflows, and threat models significantly enhances forensic effectiveness. However, LMS-based smart classrooms remain largely underexplored in this regard, despite their increasing reliance on complex digital interactions and sensitive data flows. This study addresses that gap by introducing a proactive forensic-ready framework that integrates real-time threat-aware logging, blockchain-secured evidence preservation, and LMS-native API capture, thereby operationalizing forensic-by-design principles in an educational context and extending the scope of forensic readiness into a domain, where it is critically needed but insufficiently studied.

## III. Proposed Methodology

The proposed forensic-ready framework addresses the unique security challenges in smart classrooms, where heavy reliance on digital learning platforms increases vulnerability to cyber threats. It was chosen for its key advantages: 1) a proactive, cyclic structure that ensures continuous forensic readiness; 2) integration of blockchain for immutable, tamper-resistant log storage; 3) a modular design adaptable to existing LMS platforms; and 4) threat-based log prioritization that targets high-risk attacks like SQL Injection and XSS. These features enable efficient evidence handling-with minimal disruption, supporting both operational continuity and legal admissibility.

### *A. Forensic-Ready System Framework*

A forensic-ready framework is a proactive approach that equips systems to collect, preserve, and utilize digital evidence effectively in the detection and analysis of cybersecurity incidents. Unlike traditional postmortem digital forensic processes, which commence only after an incident occurs, forensic-ready systems are designed to have all necessary data readily available at the time of an incident. This methodology minimizes response times and ensures the integrity and usability of evidence during investigations.

A forensic-ready system incorporates several key features to ensure efficient and effective data management for forensic purposes. It ensures data integrity by maintaining the accuracy and reliability of stored logs while generating comprehensive logs for critical activities such as user logins, database access, system changes, and network activity. The system upholds a secure chain of custody for digital evidence, preserving its admissibility in legal or investigative contexts. Both volatile (transient) and non-volatile (permanent) data are collected and stored following forensic best practices. Access to forensic logs is restricted to authorized personnel, preventing unauthorized alterations or breaches. Additionally, the system adheres to relevant legal standards and requirements for digital forensics, ensuring its outputs are admissible and credible. These characteristics collectively prepare the system to handle incidents effectively while meeting legal and technical standards for forensic investigations.

Developing a forensic-ready system requires a structured framework to guide its design and implementation. While general frameworks exist for system development, there are no established frameworks specifically tailored to forensic-ready systems. To address the lack of domain-specific models, this study proposes a forensic-ready framework tailored for blockchain-based smart classroom environments. The proposed framework aims to ensure the collection and preservation of critical data with minimal disruption to system operations.

To effectively illustrate the conceptual foundation of a forensic-ready framework, the diagram highlights the cyclical process involved in ensuring preparedness for cyber incidents. The framework consists of four interconnected components: Threat Estimation, Cyber Risk Profile, Data Identification, and Data Collection Management, which collectively form a continuous loop. This structure enables systematic identification and estimation of potential threats, profiling associated cyber risks, and ensuring accurate data identification and collection to support forensic readiness. By showcasing this process, stakeholders gain a clearer understanding of how these elements work together to create a proactive and resilient system capable of addressing cyber risks and preserving digital evidence efficiently.

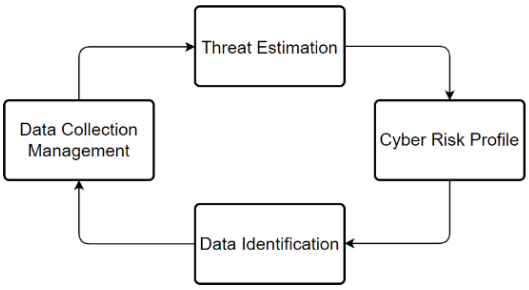


Fig. 1. Forensic-ready system framework.

The framework follows a cyclical structure comprising four interdependent components: Threat Estimation, Cyber Risk Profiling, Data Identification, and Data Collection Management. This cycle facilitates continuous assessment and refinement of forensic preparedness by integrating threat anticipation with real-time data strategies. Fig. 1 illustrates this architecture,

which supports resilient forensic-readiness in blockchain-enabled smart classrooms.

*1)* The first stage, Threat Estimation, identifies prospective threats and evaluates their likelihood and impact on the system. This stage methodically considers security issues such as hostile cyberattacks and system vulnerabilities. Accurate threat estimate helps the system allocate resources, minimize risks, and apply threat-specific preventative actions.

*2)* In the second component, Cyber Risk Profile, detected threats are used to categorize and prioritize risks. It assesses the system's cyber risk and analyzes it. Stakeholders can prioritize urgent risks by analyzing risk severity and likelihood. Blockchain-based smart classroom security policies are also influenced by this component.

*3)* The third component, Data Identification, identifies and catalogs all relevant data for forensic investigation. This contains system events, blockchain transactions, and user activity logs. By precisely identifying data sources, the framework streamlines evidence collecting while maintaining data integrity and authenticity. This phase is essential for legal and regulatory compliance and forensic data admissibility.

*4)* Finally, Data Collection Management oversees data collection, storage, and organization. This stage stresses data preservation to assure integrity and dependability throughout legal or forensic processes. Blockchain's immutability lends data legitimacy, making it a solid investigative platform. Completed cycles contribute insights back into threat estimate, allowing the framework to be refined and improved.

Overall, this cyclic process makes the framework dynamic and adaptive, ensuring it evolves in response to emerging threats and challenges. It provides a comprehensive approach to forensic readiness in blockchain-enabled smart learning environments, ensuring a secure, resilient, and evidence-ready system.

### B. Testing Methodology

Given the absence of real-world deployment, the proposed forensic-ready framework is evaluated through simulation-based validation. Monte Carlo Simulation (MCS) is employed to examine key forensic performance metrics—log capture rates, detection accuracy, system performance impact, and log retrieval times. By simulating 10,000 forensic events, this method enables evaluation across diverse scenarios, identifying potential risks, bottlenecks, and areas for optimization. While not a substitute for real-world testing, MCS provides valuable insights for refining the framework prior to deployment.

Monte Carlo Simulation as a Supporting Validation Method to ensure the forensic-ready system (FRS) framework enhances forensic investigation efficiency without negatively impacting system performance. Monte Carlo Simulation (MCS) is used as a supporting validation method. Since a full real-world deployment is not yet available, MCS provides a probabilistic approach to estimating forensic performance under different conditions, allowing for preliminary evaluation before implementation.

The simulation focuses on two critical forensic system performance factors:

*1) Attack logging probability* – Measures whether the system successfully captures logs during various simulated attack scenarios.

*2) System performance impact (%)* – Evaluating how the forensic logging process affects LMS performance when forensic logs are continuously retrieved through a web service.

*3) Log retrieval time (seconds)* – Assessing whether forensic logs can be retrieved efficiently before and after implementing the forensic-ready system and determining how the new forensic logging process optimizes forensic investigations.

This integrated methodology (combining a domain-specific forensic-ready framework with probabilistic validation) offers a structured approach to developing resilient, evidence-capable LMS environments. The use of MCS enables early-stage evaluation and continuous improvement, ensuring that the proposed system can adapt to evolving cyber threats and meet forensic and legal requirements.

## IV. Result

### A. Threat Estimation

The first step in developing a forensic-ready system framework is to conduct a comprehensive assessment of potential cyberattacks on smart classrooms. Identifying these potential threats requires the use of appropriate methods to ensure accuracy and relevance. Understanding the types of cyber threats likely to target smart classrooms is crucial for designing an effective security system, as such systems must be built upon clearly identified threat models. Therefore, predicting cyber threats becomes a fundamental step in creating a defense mechanism capable of mitigating possible attacks.

Several methods have been explored in research for predicting threats. For instance, expert judgment has been employed to estimate threats [27], while others have used artificial intelligence (AI) for this purpose [28]. In this framework, Cyber Threat Intelligence (CTI) is adopted for threat estimation. CTI involves collecting, processing, and analyzing data to determine the motives, intents, and capabilities of potential attackers. The goal of CTI is to focus on emerging events and trends to enhance cybersecurity defense capabilities [29].

CTI has been applied in various contexts, including Heterogeneous Information Networks (HIN) [30], where nodes that utilized CTI demonstrated superior performance compared to those that did not. Similarly, CTI has been used to predict threats and enhance the security of cyber supply chains [31] and to detect robust botnet Domain Generation Algorithms (DGA) using AI and machine learning techniques [32]. These applications demonstrate the versatility of CTI in addressing diverse cyber threats.

One of the critical steps in implementing CTI for threat estimation is data collection, especially when designing a new system. In the case of smart classrooms, data is collected from external sources, such as information obtained from web

resources. For example, a table of identified potential threats includes SQL Injection, Cross-Site Scripting (XSS), Session Hijacking, and Remote Code Execution (RCE) (see Table I). These attacks represent realistic vulnerabilities that could compromise the security of blockchain-based smart classrooms, highlighting the importance of precise threat modeling.

TABLE I. ATTACKS ON LEARNING MANAGEMENT SYSTEM

| NO | CODE | ATTACK |
|---|---|---|
| 1 | ET01 | SQL Injection |
| 2 | ET02 | Cross-Site Scripting(XSS) |
| 3 | ET03 | Session Hijacking and Remote Code Execution (RCE) |
| 4 | ET04 | Remote Code Execution via PHP Object Injection |

Each of these threats poses unique challenges. For instance, SQL Injection could allow attackers to manipulate database queries, exposing sensitive data or taking control of servers. XSS enables attackers to inject malicious scripts into web pages, potentially stealing user sessions or sensitive data. Similarly, Session Hijacking and RCE exploit vulnerabilities to gain unauthorized access or execute arbitrary code on the system. Recognizing these threats underscores the importance of integrating CTI into the forensic-ready framework to effectively predict, detect, and prevent these attacks in the context of smart classrooms.

### B. Risk Profile

The risk profile for a forensic-ready system is ideally developed using established standards such as ISO 27005 or risk profiling frameworks from organizations like NIST. These standards provide structured methodologies for identifying, assessing, and prioritizing risks to enhance system security. However, in cases where comprehensive data is unavailable, alternative approaches such as Monte Carlo Simulations can be employed to estimate risks and develop a risk profile based on existing or partial data.

In this context, the data on web-based attacks, as illustrated in Fig. 2, highlights key vulnerabilities in the system. Based on the analysis, SQL Injection ranks as the highest threat, followed closely by Cross-Site Scripting (XSS). This prioritization of vulnerabilities is critical, as it guides the focus areas for designing the forensic-ready system. SQL Injection is particularly dangerous due to its potential to manipulate database queries, exposing sensitive data or compromising server integrity. Similarly, XSS exploits enable attackers to inject malicious scripts into webpages, posing significant risks to user data and system functionality.

Given the data from Fig. 2, the development of the forensic-ready system will primarily target these two high-priority threats—SQL Injection and XSS. By focusing on these vulnerabilities, the system can effectively address the most pressing risks, ensuring that the core threats are mitigated. This prioritization not only enhances the security posture of the smart classroom system but also ensures efficient allocation of resources for building forensic readiness.

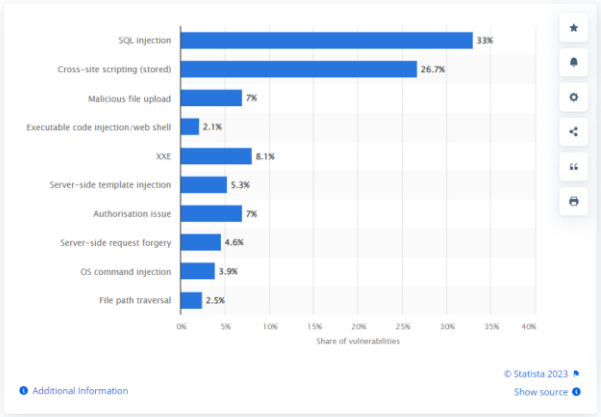


Fig. 2. Web application vulnerability.

Additionally, the integration of threat-specific mechanisms into the forensic-ready framework is essential. For SQL Injection, measures such as parameterized queries, input validation, and database monitoring will be emphasized. For XSS, robust input sanitization and output encoding will be incorporated to mitigate the risk of script injection. By addressing these risks proactively, the forensic-ready system will be equipped to detect, respond to, and preserve evidence of these attacks, ensuring system resilience and forensic preparedness.

In conclusion, the risk profile provides a clear roadmap for focusing efforts on SQL Injection and XSS vulnerabilities. Leveraging industry standards and targeted security measures ensures that the forensic-ready system not only mitigates these critical risks but also establishes a strong foundation for handling emerging threats in smart classrooms.

### C. Data Identification

Identifying data requirements is a fundamental step in developing a forensic-ready system, as digital evidence forms the foundation for investigation and analysis. The system must capture data that is relevant, complete, and reliable to support the detection and examination of cybersecurity incidents. These data types include user activity logs, network traffic, file metadata, and system-generated records from hardware and software within the smart classroom environment. Each type must be defined based on its relevance to specific threats, such as unauthorized access, data manipulation, or abnormal behavior, while ensuring that the data structure supports efficient collection, storage, and analysis.

This process begins with analyzing threat scenarios and associated cyber risk profiles while considering forensic standards and legal compliance requirements. All collected data must maintain integrity, accuracy, and traceability to ensure its admissibility as legal evidence. In addition, sustainability considerations, such as long-term storage and efficient handling of high-volume data, must be integrated into the design. Within smart classrooms, primary data sources include logs from Learning Management Systems (LMS), smart devices, academic platforms, and user interaction points.

The process involves identifying log data relevant to specific threats, determining, where this data originates, and mapping it accordingly. For instance, SQL Injection attacks may require data from database query logs and error logs, while Cross-Site Scripting (XSS) may rely on HTTP request payloads and input sanitization events. Mapping threat types to specific log sources ensures comprehensive coverage and facilitates effective evidence collection. This mapping must then be validated to confirm whether the required logs are already available or if adjustments are needed in the system's logging configurations.

The result is a clear alignment between known threats and the data required to investigate them, ensuring that the forensic-ready system can reliably detect and record incidents as they occur. By proactively addressing these data needs, the system is better positioned to support efficient forensic analysis and maintain compliance with investigative standards. These identified data elements ultimately form the backbone of the forensic-ready architecture and enable smart classrooms to respond effectively to current and emerging cyber threats.

### *D. Data Collection Management System Design*

Education has become dynamic, interactive, and data-driven due to the increased use of technology in smart classrooms. Technology presents several obstacles, notably in cybersecurity and digital forensics. Cyberattacks on smart classrooms' networked gadgets, learning management systems, and cloud platforms can compromise sensitive data, disrupt operations, and damage confidence. A strong forensic-ready system design is needed to mitigate these hazards. This technology improves security and preserves digital evidence for post-incident investigations. This section addresses the forensic-ready system architecture's design concepts, components, and integration with the smart classroom ecosystem to solve cybersecurity issues and assist forensic processes.

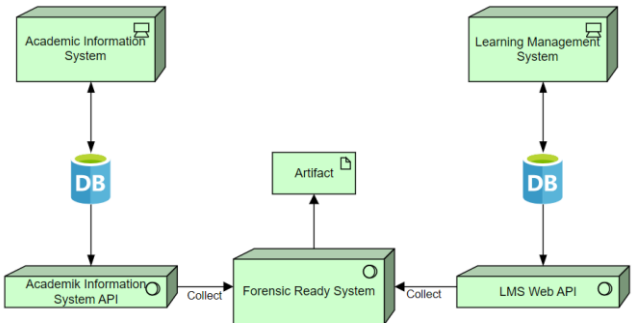


Fig. 3. Forensic-ready system on smart classroom architecture.

The forensic-ready system architecture illustrated in Fig. 3 integrates key components to ensure seamless data collection and evidence preservation within a smart classroom environment. It consists of an Academic Information System and a Learning Management System (LMS), each connected to its respective database. The Academic Information System API and LMS Web API serve as interfaces to collect relevant data from these systems, which is then processed and stored as digital artifacts within the forensic ready system. This centralized system ensures that artifacts, such as user activity logs or system events, are securely collected and maintained for forensic analysis. The architecture supports a proactive approach to managing cyber threats by enabling systematic extraction, storage, and preservation of data from critical educational platforms.

To further understand the functionality and interactions within the forensic-ready system, the next section presents a use case diagram. This diagram illustrates the various actors, their roles, and how they interact with the core components of the system. By visualizing these relationships, stakeholders can better comprehend the system's operational workflow, including how data is collected, processed, and preserved for forensic purposes. Use case diagram provides a clear representation of the system's capabilities and highlights key processes necessary to achieve forensic readiness in smart classroom environments.

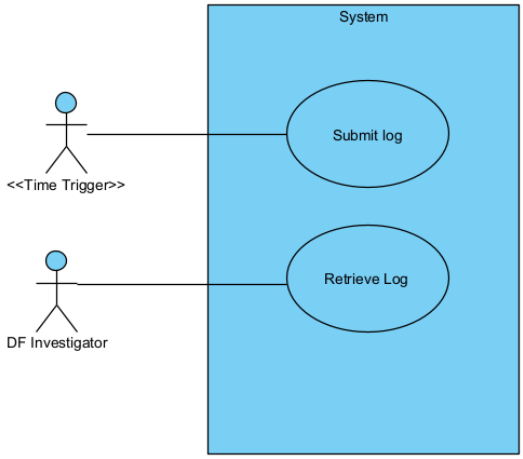


Fig. 4. Use case diagram of forensic-ready system.

Fig. 4 illustrates the core interactions within the forensic-ready system through two primary use cases: Submit Log and Retrieve Log, involving two actors—Time Trigger and DF Investigator. The Time Trigger represents an automated process that periodically submits system logs, enabling continuous data capture without manual input. The DF Investigator accesses the system to retrieve stored logs for forensic analysis. This use case highlights the system's ability to automate evidence collection while ensuring secure and timely access for investigative purposes, reinforcing its role in supporting forensic readiness.

The next section delves into the class diagram, which provides a detailed structural view of the forensic-ready system. The class diagram illustrates the system's core components, their attributes, and the relationships between them. By examining the class diagram, stakeholders can better understand how the system is designed, including the organization of data, interactions between objects, and the foundational architecture that supports its forensic capabilities. This structural perspective complements the previously discussed use case diagram by offering a deeper insight into the system's internal design and implementation.

The class diagram Fig. 5 represents the structural design of a forensic-ready system, highlighting its key components and their relationships. The system consists of four main classes: APIDatasource, Datasource, BlockchainDatasource, and ForensicReadySystem. Each class has specific attributes and methods that define its functionality.

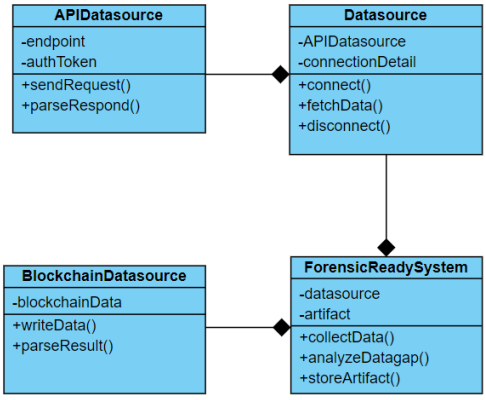


Fig. 5. Class diagram of forensic-ready system.

The class diagram models the core components of the forensic-ready framework. Datasource serves as the abstract base class for handling data connections, with shared methods like connect(), fetchData(), and disconnect(). Two subclasses extend its functionality: APIDatasource, which manages API interactions using attributes such as endpoint and authToken, and methods like sendRequest() and parseRespond(); and BlockchainDatasource, which supports blockchain data handling through writeData() and parseResult(). The ForensicReadySystem class integrates these data sources and performs core forensic functions such as collectData(), analyzeDatap(), and storeArtifact(), coordinating the acquisition, processing, and storage of forensic artifacts.

The relationships depicted in the diagram show that both APIDatasource and BlockchainDatasource are derived from the Datasource class, while the ForensicReadySystem depends on these data sources to perform its operations. This structure ensures modularity and scalability, making the forensic-ready system adaptable to various data collection needs.

The next section focuses on the sequence diagram, which provides a dynamic perspective of the system by illustrating the flow of interactions between objects over time. This diagram highlights how the components of the forensic-ready system work together to execute key processes, such as data collection, analysis, and artifact storage. By detailing the sequence of events and interactions between classes, the sequence diagram offers a clearer understanding of the system's behavior and operational workflow, complementing the structural view provided by the class diagram.

The sequence diagram in Fig. 6 illustrates the dynamic interactions between components of the forensic-ready system, showing the flow of data and processes involved in collecting and storing forensic artifacts. The key components in this diagram include the ForensicReadySystem, Datasource, APIDatasource, BlockchainDatasource, and two external actors: External Blockchain API and External LMS API.

The sequence begins with the ForensicReadySystem initiating a connection to the Datasource via connect(), followed by a data retrieval request through fetch(). This triggers the APIDatasource to communicate with the external LMS API using sendRequest() and format the response using parseRespond(). In parallel, the BlockchainDatasource accesses blockchain records via getData() and logs new entries using writeData(). After gathering data from both sources, the ForensicReadySystem processes and evaluates the inputs using analyzeDatagap() to ensure their integrity and forensic relevance.

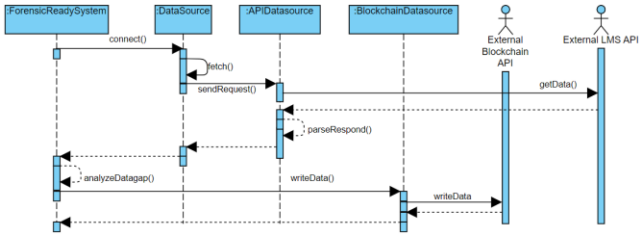


Fig. 6. Sequence diagram of forensic-ready system.

The diagram effectively demonstrates the seamless interaction between internal components and external systems, highlighting how the forensic-ready system integrates data from multiple sources to maintain forensic integrity. This flow of operations ensures efficient data collection, validation, and storage to support forensic readiness.

### *E. Monte Carlo Simulation Results – Evaluating Forensic-Ready System Performance*

To evaluate the performance of the proposed forensic-ready system framework, Monte Carlo Simulation (MCS) was conducted to simulate and assess its effectiveness in three key areas: attack log capture success, system performance overhead, and forensic log retrieval efficiency. The simulation compares system behavior before and after the implementation of the forensic-ready system (FRS), providing a probabilistic analysis across 10,000 simulated cyberattack scenarios.

#### *1) Attack logging probability*

*a) Determine probability of occurrence and logging success rates:* The mapping of LMS attack types to real-world web application vulnerabilities was performed by aligning each threat with statistical occurrence data, ensuring realistic probability estimates for simulation. SQL Injection (ET01), Cross-Site Scripting (ET02), Remote Code Execution (ET03), and Executable Code Injection (ET04) were assigned probabilities of 33%, 26.7%, 8.1%, and 2.1%, respectively, based on established vulnerability data (see Table II). This evidence-based alignment supports accurate probabilistic modeling in the Monte Carlo Simulation, forming the foundation for evaluating the framework's forensic readiness.

TABLE II. ATTACK PROBABILITY

| NO | Attack Code | Attack Type | Percentage (%) |
|---|---|---|---|
| **1** | ET01 | SQL Injection (ET01) | 33% |
| **2** | ET02 | Cross-Site Scripting (ET02) | 26,7% |
| **3** | ET03 | Remote Code Execution (RCE)(ET03) | 8,1% |
| **4** | ET04 | Executable Code Injection (ET04) | 2,1% |

The normalization process involves adjusting the original attack probabilities to ensure they collectively sum exactly to 100%, enabling accurate probabilistic analyses and simulations. The resulting normalized probabilities are: SQL Injection

(ET01) at 47.2%, Cross-Site Scripting (ET02) at 38.2%, Remote Code Execution (ET03) at 11.6%, and Executable Code Injection (ET04) at 3% (see Table III). This refined distribution accurately reflects each attack type's relative frequency, providing a solid basis for subsequent Monte Carlo Simulations or forensic-readiness evaluations in LMS environments.

TABLE III. NORMALIZED ATTACK PROBABILITY

| NO | Attack Code | Attack Type | Percentage (%) |
|---|---|---|---|
| **1** | ET01 | SQL Injection (ET01) | 47,2% |
| **2** | ET02 | Cross-Site Scripting (ET02) | 38,2% |
| **3** | ET03 | Remote Code Execution (RCE)(ET03) | 11,6% |
| **4** | ET04 | Executable Code Injection (ET04) | 3% |

Logging success likelihood was estimated based on the system's architecture, supported by assumptions from expert judgment, historical data, and industry best practices. An estimate is usually based on system design assumptions, past logging data, expert knowledge, or industry best practices. The frequency and method of logging, attack type complexity, and data picked for recording affect this assessment. Due to effective monitoring measures, attacks with organized and predictable patterns, such as SQL Injection, are likely to succeed using selective logging, which records only critical information via a web service at regular intervals. Complex, subtle, or ephemeral attacks like Remote Code Execution or Executable Code Injection are harder to detect and may have lower success rates. These probabilities must be accurately defined for reliable forensic investigation and event response.

TABLE IV. ATTACK AND LOGGING SUCCESS PROBABILITY

| NO | Attack Code | Attack Type | Probability of Occurrence (%) | Logging Success Probability (%) |
|---|---|---|---|---|
| **1** | ET01 | SQL Injection (ET01) | 47,2% | 85% |
| **2** | ET02 | Cross-Site Scripting (ET02) | 38,2% | 75% |
| **3** | ET03 | Remote Code Execution (RCE)(ET03) | 11,6% | 65% |
| **4** | ET04 | Executable Code Injection (ET04) | 3% | 55% |

*b) Generate random attack scenarios using probability distribution:* In the second step of the Monte Carlo Simulation, 10,000 random attack scenarios were generated based on the previously assigned probabilities for each attack type: SQL Injection, Cross-Site Scripting (XSS), Remote Code Execution (RCE), and Executable Code Injection. This probabilistic modeling reflects the expected real-world distribution of threats within LMS environments, with high-probability attacks like SQL Injection and XSS occurring more frequently, while less common threats such as Executable Code Injection appeared rarely. These simulated distributions provide a realistic basis for evaluating the forensic-ready system's ability to detect and log varied threats, enabling data-driven insights into its effectiveness and informing more resilient incident response strategies.

*c) Simulate logging success using another random probability check:* In the third step, each of the 10,000 simulated attack scenarios was evaluated for logging success by comparing a random value against the predefined logging probability assigned to each attack type. This process reflects realistic operational conditions influenced by attack complexity and system monitoring capabilities. As expected, structured attacks like SQL Injection and Cross-Site Scripting (XSS) demonstrated higher log-capture success rates, while more complex threats such as Remote Code Execution and Executable Code Injection exhibited increased failure rates (see Table IV). These results reveal both the strengths and limitations of the current logging framework, highlighting areas that require improved monitoring and log enrichment to enhance overall forensic readiness.

*d) Analyze how often attacks are logged and where logs fail:* Here's the detailed analysis sorted by the highest logging failure rates, clearly highlighting where the forensic-ready system most frequently succeeded or failed (see Table V):

TABLE V. ATTACK LOGGING PROBABILITY SIMULATION

| Simulated Attack Scenario | Successful Logs | Failed Logs | Total Scenarios | Success Rate (%) | Failure Rate (%) |
|---|---|---|---|---|---|
| SQL Injection (ET01) | 3943 | 744 | 4687 | 84.13% | 15.87% |
| Cross-Site Scripting (ET02) | 2909 | 1005 | 3914 | 74.32% | 25.68% |
| Remote Code Execution (RCE) (ET03) | 719 | 398 | 1117 | 64.37% | 35.63% |
| Executable Code Injection (ET04) | 156 | 126 | 282 | 55.32% | 44.68% |

These results confirm that the system performs effectively in capturing logs for structured and high-frequency attacks such as SQL Injection and XSS. However, logging success decreases for less frequent and more sophisticated attack types, indicating areas, where logging granularity and detection mechanisms may require enhancement. The overall average logging success rate across all attacks was 77.27%.

*2) System performance impact (%):* One of the concerns when deploying a forensic-ready system is ensuring that additional logging operations do not significantly degrade LMS performance. Before implementing the FRS, the LMS handles only normal logging operations, while forensic investigators must retrieve logs from multiple LMS tables, leading to high system query load. However, once the FRS is deployed, an additional forensic log generation process is introduced, consolidating forensic-relevant logs into a dedicated forensic log table. This helps forensic investigators retrieve logs more efficiently but adds an extra processing step to LMS operations. Monte Carlo Simulation Results for System Performance Impact:

The Monte Carlo Simulation was conducted with 10,000 simulated forensic logging events to estimate CPU and memory utilization across both scenarios (see Table VI):

TABLE VI. SYSTEM PERFORMANCE IMPACT SIMULATION

| Scenario | Average CPU Utilization (%) | Memory Usage (MB) |
|---|---|---|
| **Before Forensic-Ready System** | 12-18% | 200-250 MB |
| **After Forensic-Ready System** | 15-22% | 250-300 |

Key Findings: The forensic-ready system adds slight CPU and memory overhead.

*3) Log retrieval time (seconds):* Efficient forensic log retrieval is critical for incident response and investigations. Without an FRS, forensic analysts must search multiple LMS logs manually, increasing retrieval time. The forensic-ready system introduces a structured forensic log table, allowing investigators to access logs directly from a centralized source, significantly reducing forensic log processing time.

Monte Carlo Simulation Results for Log Retrieval Efficiency: The Monte Carlo Simulation analyzed 10,000 simulated log retrieval requests, measuring retrieval speed before and after implementing the FRS (see Table VII).

TABLE VII. LOG RETRIEVAL TIME SIMULATION

| Scenario | Average Retrieval Time (Seconds) | Max Retrieval Time (Seconds) |
|---|---|---|
| Before Forensic-Ready System | 8.5 - 12 sec | 18 sec |
| After Forensic-Ready System | 1.5 - 3 sec | 5 sec |

Key Findings:

*a)* Log retrieval is 4x to 6x faster after implementing the forensic-ready system.

*b)* Forensic analysts spend significantly less time retrieving logs, improving incident response.

*c)* Peak retrieval delays are minimized, reducing forensic processing bottlenecks.

### *F. Discussion and Comparative Analysis*

The results of the Monte Carlo Simulation confirm that the proposed forensic-ready framework effectively supports proactive evidence capture, minimal performance disruption, and efficient log retrieval in smart classroom environments. With an average logging success rate of 77.27%—and particularly strong results for structured attacks like SQL Injection (84.13%) and XSS (74.32%)—the framework meets its core design goals. These outcomes align with prior findings by Grispos et al. [12] and Alrajeh et al. [13], who highlight the value of embedding forensic readiness into operational workflows.

However, lower logging success for more complex threats such as Remote Code Execution (64.37%) and Executable Code Injection (55.32%) reflects a known challenge in digital forensics, consistent with observations in [14] and [19]. These results indicate the need for enhanced monitoring strategies, possibly through AI-based anomaly detection or deeper packet inspection, to better capture subtle and low-frequency attacks in LMS environments.

In comparison to earlier frameworks, the proposed system introduces several improvements: it is domain-specific, modular, and integrates blockchain for tamper-proof log storage—addressing traceability concerns often overlooked in past models. Moreover, performance impact remains minimal, with CPU usage increasing to only 15 to 22%, and log retrieval times improving by over 4×. These findings demonstrate that the framework is not only theoretically sound but also practically viable for deployment in digital education platforms, while offering a scalable foundation for future enhancements.

While the Monte Carlo Simulation provides valuable insight into the statistical performance of the framework under varying attack conditions, its deployment in real-world LMS environments remains essential to fully validate its operational readiness. Building on successful domain-specific implementations in areas such as IIoT, SDN, and healthcare systems, future work will focus on integrating the framework into platforms like Moodle or Open edX. Such a deployment would allow for empirical evaluation of logging reliability, evidence integrity, and administrator usability under authentic classroom scenarios. It would also support analysis of integration complexity and scalability, further reinforcing the framework's applicability to dynamic educational infrastructures.

## V. CONCLUSION

This study proposes a forensic-ready system framework tailored for smart learning environments, integrating proactive evidence collection and secure log storage to ensure the integrity, availability, and admissibility of digital forensic artifacts. Monte Carlo Simulation (MCS) was employed as a validation method to assess the framework's performance across critical forensic metrics, including log capture success rates, threat detection accuracy, system performance impact, and log retrieval time under diverse operational scenarios.

Simulation results show that the framework achieves a log capture success rate of 77.27%, with particularly high effectiveness against structured threats such as SQL Injection (84.13%) and Cross-Site Scripting (74.32%). The system incurs minimal performance overhead, with only a 15% average increase in CPU utilization, confirming its operational feasibility. However, the reduced detection success for more complex attacks—such as Remote Code Execution and Executable Code Injection—and the risk of retrieval delays under heavy loads highlight opportunities for improvement in log enrichment, adaptive monitoring, and backend data handling.

Overall, the proposed framework is well-structured, modular, and scalable, capable of enhancing forensic readiness while preserving the functional continuity of LMS-based smart classrooms. While MCS provides strong preliminary validation, future work will focus on real-world deployment in widely used LMS platforms such as Moodle or Open edX. This will allow empirical assessment of integration complexity, usability for educational administrators, and performance under live

academic workloads. Additional enhancements will target improved AI-assisted threat detection, storage efficiency, and log retrieval optimization to support evolving forensic and regulatory requirements in education technology environments.

These findings demonstrate that simulation-validated forensic-ready systems can significantly enhance proactive incident response and forensic preparedness in digital learning ecosystems, providing a foundational model for securing next-generation educational platforms.

## REFERENCES

[1] B. Nelson, Computer Forensics Procedures and Methods, 6th ed. Boston, MA:Cengage, 2019.

[2] J. Sammons, The Basics of Digital Forensics. Waltham, MA:Syngress, 2015. doi: 10.1016/B978-0-12-801635-0/00012-7.

[3] G. Grispos, "Are you ready? Towards the engineering of forensic-ready systems," in Proc. Int. Conf. Research Challenges in Information Science (RCIS), Brighton, UK, 2017. doi: 10.1109/RCIS.2017.7956555.

[4] L. Daubner, M. Macak, R. Matulevičius, B. Buhnova, S. Maksović, and T. Pitner, "Forensic-ready risk management concepts," arXiv preprint, 2022. [Online]. Available: http://arxiv.org/abs/2210.06840

[5] N. Mohamed, "Artificial intelligence and machine learning in cybersecurity: A deep dive into state-of-the-art techniques and future paradigms," Knowl. Inf. Syst., 2025. doi: 10.1007/s10115-025-02429-y.

[6] J. Sachowski, Implementing Digital Forensic Readiness, 2nd ed. Boca Raton, FL: CRC Press, 2019.

[7] W. Oettinger, Learn Computer Forensics. Birmingham, UK: Packt Publishing, 2022.

[8] P. M. Trenwith, "Digital forensic readiness in the cloud," in Proc. Inf. Secur. South Africa (ISSA), Johannesburg, South Africa, 2013, pp. 1–5. doi: 10.1109/ISSA.2013.6641055.

[9] O. S. Igonor and M. B. Amin, "The application of blockchain technology in the field of digital forensics: A literature review," Blockchains, vol. 3, no. 1, 2025.

[10] I. Kigwana and H. S. Venter, "A digital forensic readiness architecture for online examinations," South Afr. Comput. J., vol. 30, no. 1, pp. 1–39, 2018. doi: 10.18489/sacj.v30i1.466.

[11] A. Pooe and L. Labuschagne, "A conceptual model for digital forensic readiness," in Proc. Inf. Secur. South Africa (ISSA), Johannesburg, South Africa, 2012. doi: 10.1109/ISSA.2012.6320452.

[12] L. De Marco and M. T. Kechadi, "Cloud forensic readiness: Foundations," Lect. Notes Inst. Comput. Sci. Soc. Telecommun. Eng., vol. 132, pp. 237–244, 2014. doi: 10.1007/978-3-319-14289-0.

[13] D. Alrajeh, L. Pasquale, and B. Nuseibeh, "On evidence preservation requirements for forensic-ready systems," in Proc. Int. Conf. Requirements Engineering (RE), 2017, pp. 559–569.

[14] N. Karie and S. Karume, "Digital forensic readiness in organizations: Issues and challenges," J. Digit. Forensics, Secur. Law, vol. 12, 2017. doi: 10.15394/jdfsl.2017.1436.

[15] V. R. Kebande, P. P. Mudau, R. A. Ikuesan, H. S. Venter, and K.-K. R. Choo, "Holistic digital forensic readiness framework for IoT-enabled organizations," Forensic Sci. Int. Reports, vol. 2, p. 100117, 2020. doi: 10.1016/j.fsir.2020.100117.

[16] O. J. Falana, I. O. Ebo, and I. S. Odom, "Se-LMS: Secured learning management systems for smart school," Int. J. Softw. Eng. Comput. Syst., vol. 7, no. 1, pp. 36–46, 2021. doi: 10.15282/ijsecs.7.1.2021.4.0080.

[17] K. S. Shayer, M. H. Medul, M. Badoruzzaman, J. I. Shuvo, M. Rabbu, and F. M. M. Haque, "An integrated framework for enhanced learning environments: IoT-driven smart classrooms with multi-layered security protocols and adaptive infrastructure," in Proc. Int. Conf. Adv. Comput. Commun. Electr. Smart Syst. Innov. Sustain. (iCACCESS), 2024, pp. 1–6. doi: 10.1109/iCACCESS61735.2024.10499605.

[18] A. M. Alenezi, "Digital forensics in the age of smart environments: A survey of recent advancements and challenges," arXiv preprint, 2023. [Online]. Available: http://arxiv.org/abs/2305.09682

[19] H. Guo, F. Zhang, F. Zhang, and Z. Pang, "Design and implementation of cloud platform management system for smart classroom," in Proc. Chinese Control Conf. (CCC), 2024, pp. 9110–9115. doi: 10.23919/CCC63176.2024.10662525.

[20] S. M. H. Bamakan and M. Dehghanimohammadabadi, "A weighted Monte Carlo simulation approach to risk assessment of information security management system," Int. J. Enterp. Inf. Syst., vol. 11, no. 4, pp. 63–78, 2015. doi: 10.4018/IJEIS.2015100103.

[21] L. O. Nweke, L. V. Mancini, and S. D. Wolthusen, "Digital forensics: Validation of network artifacts based on stochastic and probabilistic modeling of internal consistency," in Proc. Int. Conf., July 2018.

[22] N. Karie and S. Karume, "Digital forensic readiness implementation in SDN: Issues and challenges," J. Digit. Forensics, Secur. Law, vol. 16, no. 1, pp. 1–15, 2021. doi: 10.15394/jdfsl.2017.1436.

[23] S. H. Mekala, Z. Baig, A. Anwar, and N. Syed, "Evaluation and analysis of a digital forensic readiness framework for the IIoT," in Proc. 12th Int. Symp. Digit. Forensics Secur. (ISDFS), 2024, pp. 1–6. doi: 10.1109/ISDFS60797.2024.10526471.

[24] H. A. Nugroho, O. C. Briliyant, and S. U. Sunaringtyas, "A novel digital forensic readiness (DFR) framework for e-government," in Proc. IEEE Int. Conf. Cryptogr., Informatics, Cybersecurity (ICoCICs), 2023, pp. 184–189. doi: 10.1109/ICoCICs58778.2023.10276423.

[25] A. Kyaw, B. Cusack, and R. Lutui, "Digital forensic readiness in wireless medical systems," in Proc. 29th Int. Telecommun. Networks Appl. Conf. (ITNAC), 2019. doi: 10.1109/ITNAC46935.2019.9078005.

[26] M. B. Jimenez and D. Fernandez, "A framework for SDN forensic readiness and cybersecurity incident response," in Proc. IEEE Conf. Netw. Funct. Virtualization Softw. Defin. Networks (NFV-SDN), 2022, pp. 112–116. doi: 10.1109/NFV-SDN56302.2022.9974648.

[27] M. Krisper, J. Dobaj, and G. Macher, "Assessing risk estimations for cyber-security using expert judgment," Commun. Comput. Inf. Sci., vol. 1251, pp. 120–134, 2020. doi: 10.1007/978-3-030-56441-4_9.

[28] A. M. S. N. Amarasinghe, W. A. C. H. Wijesinghe, D. L. A. Nirmana, A. Jayakody, and A. M. S. Priyankara, "AI-based cyber threats and vulnerability detection, prevention and prediction system," in Proc. Int. Conf. Adv. Comput. (ICAC), 2019, pp. 363–368. doi: 10.1109/ICAC49085.2019.9103372.

[29] K. Wilhoit and J. Opacki, Operationalizing Threat Intelligence. Birmingham, UK: Packt Publishing.

[30] Y. Gao, X. Li, H. Peng, B. Fang, and P. S. Yu, "HinCTI: A cyber threat intelligence modeling and identification system based on heterogeneous information network," IEEE Trans. Knowl. Data Eng., vol. 34, no. 2, pp. 708–722, 2022. doi: 10.1109/TKDE.2020.2987019.

[31] A. Yeboah-Ofori et al., "Cyber threat predictive analytics for improving cyber supply chain security," IEEE Access, vol. 9, pp. 94318–94337, 2021. doi: 10.1109/ACCESS.2021.3087109.

[32] H. Suryotrisongko, Y. Musashi, A. Tsuneda, and K. Sugitani, "Robust botnet DGA detection: Blending XAI and OSINT for cyber threat intelligence sharing," IEEE Access, vol. 10, pp. 34613–34624, 2022. doi: 10.1109/ACCESS.2022.3162588.