Volume 4 Issue VII, August 2025, No. 69, pp. 878-893
Submitted 22/4/2025; Final peer review 8/7/2025
Online Publication 2/8/2025
Available Online at http://www.ijortacs.com

DEVELOPMENT OF A REAL-TIME VULNERABILITY MANAGEMENT SYSTEM FOR HEALTHCARE CYBER-PHYSICAL SYSTEMS USING A HYBRID DEEP LEARNING

[1*]Mba Chioma Juliet, [1]Amadi Gloria Ebere, [2]Ezugwu Lilian Martina
[1*]Enugu State Polytechnic Iwollo
[1]Enugu State University Science and Technology
[2]Enugu State College of Education (Technical), Enugu
Email: [1*]chiomajulietmba@gmail.com; [1]ebbyglory@yahoo.com [2]lilianezugwu6@gmail.com

Abstract

The growth of the utilization of Internet of Things (IoT) and cyber-physical systems (CPS) in the healthcare sector has come along with new vulnerabilities in cybersecurity, which endanger the safety and integrity of patients and data. Considering that a hybrid deep learning framework based on Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) networks is developed in the context of this study, this paper introduces real-time vulnerability management system of medical cyber-physical systems on the hybrid deep learning framework. The system was tested and trained based on the data of vulnerabilities taken at the University of Nigeria Teaching Hospital (UNTH), one of the targeted intensive care units (ICU) devices between 2019 and 2022. The data was also extensively pre-processed (number of missing values was imputed, the normalization and class balancing with random under-sampling was performed). The CNN layer allowed extracting spatial features whereas the LSTM layer allowed capturing temporal patterns in network traffic and system logs. It was written in Python, TensorFlow, and Keras with tools that allow real-time scanning included, namely ClamAV and Nmap. Evaluation of the performance has shown high precision (97 training and 93 validation), and the F1-score was 0.94 due to the good result of the system in identifying and categorizing the known and upcoming vulnerabilities. The system was found to be real-time deployable as well as providing proactive threat detection at levels of multiple networks. The study would help enhance cybersecurity resilience in healthcare, as the method relied on by this research allowed early warning against and remediation of risks within mission-critical medical settings.

Keywords: **Healthcare; Internet of Things; Vulnerability; Deep Learning; CNN; LSTM**

1. INTRODUCTION

Cyber-Physical Systems (CPS) is foundational to critical sectors such as smart grids, industrial automation, healthcare, and transportation system (Parades et al., 2024). These systems are becoming increasingly complex as they integrate vast networks of physical components, computational units, and communication infrastructure.CPS represent a convergence of the physical and cyber worlds, integrating computational elements with physical processes to create systems that can monitor, control, and optimize operations in real-time (Yuan et al., 2024). CPS consists of interconnected

components that communicate and collaborate to achieve specific objectives, often in complex environments. For instance, in healthcare, CPS can include devices like pacemakers, insulin pumps, and telehealth systems, which collect and analyze patient data, providing real-time insights for better clinical decision-making (Abdelrahman et al., 2023).

The key characteristic of CPS is their ability to interact with the physical environment through sensors and actuators while utilizing advanced computational algorithms to process and analyze data. CPS consists of three interconnected layers that work together to ensure efficient communication, processing, and control of physical processes: the physical layer, the network layer, and the application layer (Chu et al., 2020). Each of these layers plays a vital role in the functioning of CPS, enabling the interaction between physical devices, computational systems, and communication networks (Bahaa et al., 2022). Understanding these layers is crucial for optimizing CPS performance, improving its security, and ensuring resilience against potential threats.

Healthcare Cyber-Physical Systems (CPS) represent the integration of physical healthcare devices and digital systems, allowing for enhanced monitoring, control, and data analysis in real-time (Carreras-Guzmanet al., 2020). These systems can include medical devices such as wearable health monitors, robotic surgical systems, and telemedicine platforms that collect and transmit patient data to healthcare providers. The interconnectivity of these devices facilitates improved patient care by enabling continuous monitoring of vital signs, remote consultations, and timely interventions (Bellman et al., 2020). However, the reliance on technology also introduces various vulnerabilities, including risks associated with data breaches, unauthorized access, and system malfunctions, which can have serious consequences for patient safety and privacy (Oks et al., 2019). Understanding the structure and functionality of healthcare CPS is essential for identifying potential weaknesses and implementing effective security measures.

Vulnerability in Cyber-Physical Systems (CPS) refers to weaknesses or flaws in the system's architecture, hardware, software, or communication networks that can be exploited by cyber attackers, resulting in potentially severe consequences for the physical processes controlled by the system (Su et al., 2024). CPS integrates both physical and digital components, making them more complex and prone to unique vulnerabilities that go beyond traditional IT systems (Tang et al., 2023). These vulnerabilities can affect various aspects of the system, such as its ability to monitor, control, and respond to real-world environments in sectors like healthcare, manufacturing, and transportation.

Effective vulnerability management in Cyber-Physical Systems involves a systematic approach to identifying, assessing, and mitigating vulnerabilities to enhance system security and resilience.Vulnerability management refers to the process of identifying, classifying, and reporting this security vulnerability in the systems (Northern et al., 2021). In the H-CPS context, vulnerability detection and control aims to identify flaws in the different service layers and report to help mitigate cyber-attacks. Currently, to successfully manage

vulnerability in H-CPS is very difficult due to the complex nature of its architecture. According to the Industrial Control System Cyber Emergency Response Team (ICS-CERT), a surge in security breaches is affecting CPS and embedded systems, thus necessitating alternative solutions in the scientific community to help manage this problem (Knowles et al., 2015; Bernieri et al., 2018).

Deep learning has revolutionized vulnerability management in Cyber-Physical Systems (CPS) by enabling advanced analytical capabilities that enhance the detection and mitigation of potential threats. Utilizing multilayered neural networks, deep learning algorithms can process vast amounts of data generated by CPS components, such as sensors and actuators, to identify patterns and anomalies indicative of vulnerabilities (Qu et al., 2023). These algorithms can be trained on historical data to recognize normal operational behaviour, allowing them to flag deviations that may signify a cyberattack or system failure. For instance, in healthcare CPS, deep learning can be employed to monitor patient monitoring devices in real-time, detecting anomalies that could suggest tampering or malfunction (Amulya et al., 2024). This proactive approach significantly enhances the ability to identify vulnerabilities before they can be exploited, thereby increasing the resilience of CPS.

In the past, basic security frameworks like intrusion detection systems, firewalls, and patch management systems were some of the popular approaches for vulnerability management, but despite their success, they were often reactive and unable to provide real-time security assurance for general CPSs. Recently, Deep Neural Networks (DNN) have resonated as a powerful tool for real-time data analysis of complex patterns, making them suitable for pro-active vulnerability detection and control in CPS (Khazraei et al., 2022; Ashraf et al., 2022); However, there is limited work on the application of Deep Leaning (DL) for vulnerability management in CPS, thus necessitating the need for this study. This work presents a real-time vulnerability management model for electronic healthcare cyber-physical system using deep neural network

2. PROPOSED SYSTEM REAL-TIME VULNERABILITY MANAGEMENT MODEL

The proposed system is designed to enhance cybersecurity in medical cyber-physical systems through a structured approach that involves multiple components. It will begin with data collection from the various layers of the medical cyber-physical system, ensuring comprehensive input from the operational environment. This data will undergo processing to clean and organize it for analysis, followed by data extraction to identify relevant features essential for vulnerability assessment. The core of the system will involve deep neural networks, where the model will be trained on the extracted features to learn patterns associated with potential vulnerabilities. Rigorous evaluation will be conducted to assess the model's performance, followed by model generation tailored for vulnerability assessment. Finally, the system will be deployed in the operational environment, allowing for real-time monitoring and

Corresponding Author Tel: +234-8039490620

proactive identification of security threats, thereby improving the resilience of medical cyber-physical systems against cyber-attacks. The Figure 1 presents the system block diagram.

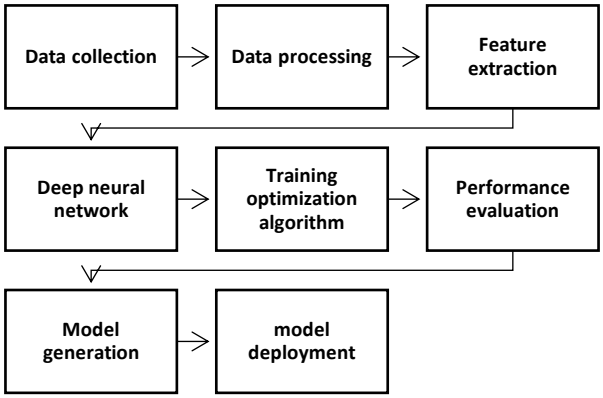

Figure 1: Block diagram of the real-time vulnerability management model.

The block diagram of the real-time vulnerability management model was presented in Figure 1. Data of common vulnerabilities in the Intensive Care Unit (ICU) of University of Nigeria Teaching Hospital (UNTH) Enugu, capturing key security attributes across the transport, network, and application layers of connected medical devices from 2019 to 2022 will be collected from a medical cyber physical system considering the different layers of the architecture. The collected data will be processed using imputation techniques and normalization approach, then feature extraction will be applied to the data, before feeding a deep neural network and then train with optimization algorithm. The performance will be evaluated using accuracy, F1-score, recall and precision and the model generated for deployment as a real-time vulnerability management model. The Figure 2 presents the component interaction diagram of the proposed system.

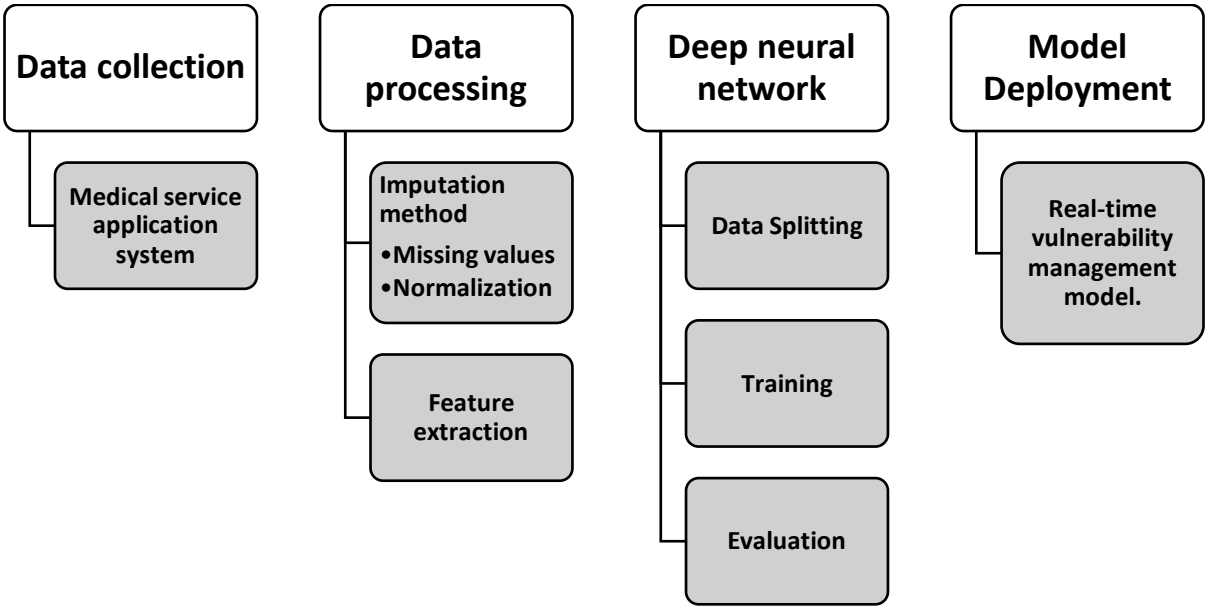

Figure 2: Component diagram of the proposed real-time vulnerability management model.

Figure 2 presents the component interaction diagram of the proposed system. Data will be collected from medical service application system. The data will be processed through imputation to remove missing values and also normalize the data through dimensionality

reduction approach and then split to train a deep neural network. During the training process, the model performance will be evaluated and the result deployed as a real-time vulnerability management model.

2.1 Data Collection

The data used for this work was collected from the University of Nigeria Teaching hospital, Enugu state Nigeria as the primary data source. The data include common vulnerabilities in the Intensive Care Unit (ICU) of the hospital, capturing key security attributes across the transport, network, and application layers of connected medical devices from 2019 to 2022. It includes structured records of known vulnerabilities, identified by `CVE ID` and categorized under `CWE ID` to specify the weakness type. Each entry provides details such as ` of Exploits` to indicate exploitation likelihood, `Vulnerability Type(s)`, and `Publish Date` with `Update Date` for tracking disclosure timelines. The `Score` (CVSS) quantifies severity, while `Gained Access Level`, `Access Complexity`, and `Authentication` highlight exploit difficulty. The dataset further assesses security impact through `Confidentiality Impact`, `Integrity Impact`, and `Availability Impact`, mapping risks to patient data privacy, device reliability, and system uptime. This comprehensive structure enables predictive modelling for risk assessment and proactive mitigation of cybersecurity threats in critical healthcare infrastructures. The sample size of the data collected is 107606 features of vulnerability. The Table 1 presents the data description.

Table 1: Data description of Healthcare IoT vulnerability

Attribute	Data Type	Description
CVE ID	String	Unique identifier for the vulnerability (e.g., CVE-2023-XXXX).
CWE ID	String	Common Weakness Enumeration (CWE) identifier for the vulnerability type.
of Exploits	Integer	Number of known exploits available for the vulnerability.
Vulnerability Type(s)	String	Type of vulnerability (e.g., SQL Injection, Buffer Overflow).
Publish Date	Date (YYYY-MM-DD)	The date when the vulnerability was publicly disclosed.
Update Date	Date (YYYY-MM-DD)	The most recent update date of the vulnerability record.
Score	Float (0.0 - 10.0)	CVSS (Common Vulnerability Scoring System) score indicating severity.
Gained Access Level	String	The level of access gained if exploited (e.g., Admin, User).
Access Complexity	String (Low/Med/High)	Difficulty of exploiting the vulnerability.
Authentication	String (Required/Not Required)	Whether authentication is needed for exploitation.
Confidentiality	String	Impact on data confidentiality if exploited.

Corresponding Author Tel: +234-8039490620

Impact	(Low/Med/High)	
Integrity Impact	String (Low/Med/High)	Impact on data integrity if exploited.
Availability Impact	String (Low/Med/High)	Impact on system availability if exploited.
Description	String	A brief summary of the vulnerability and its impact.

2.2 Data Preparation

The collected data were processed using visualization, normalization and balancing approach. First the data structure was visualized in excel form to check for missing and duplicate values. This was done carefully using manual physical inspection by the researcher. The outcome showed that the data has no duplicate and missing values, and in addition all the features were observed to be numeric apart from the unique identifier. Data normalization was applied for dimensionality reduction using resampling approach based on Min-Max scaler technique as shown in Equation 1 (Khalid et al., 2024).

$$X_{i,j} = (X_{i,j} - Min(X_j))/(Max(X_j) - Min(X_j)) \quad (1)$$

Where Xj represents the features of the j_{th} credit card transactions, $X_{i,j}$ is the features X_j of sample i. for the data distribution of target variables.

The imbalance data structure prompt the need for class balancing. This was address by adopting the random under sampling approach in Khalid et al. (2024). The algorithm was presented using the relationship between fraud transaction subset defined as D^1, legitimate subset defined as D^0 and D^' which represents under sample dataset.

Algorithm 1: Random class under sampling algorithm

$Input: D = DD^o \; U \; D^1$
$D' = D^1$
$For \; j = 1,2, \ldots \ldots \ldots card \; D^1 \; do$
$Select \; random \; sample \; x \in D^0$
$D' = D'U \{x\}$
$Delete \; x \; from \; D^0$
$End \; for$
$Output \; D'$

3. CNN-LSTM MODEL

A CNN-LSTM model is a hybrid deep learning architecture that combines CNN for spatial feature extraction and LSTM networks for capturing temporal dependencies in sequential data. CNN layers first process raw data, identifying key spatial patterns, while LSTM layers analyze the extracted features over time to detect trends, anomalies, or cyber threats. This model is highly effective for intrusion detection, vulnerability assessment, and anomaly detection in healthcare systems, enabling real-time security monitoring by identifying sequential attack patterns and abnormal behaviours in connected medical devices. Figure 1 presents the flow chart of the CNN+LSTM.

Corresponding Author Tel: +234-8039490620

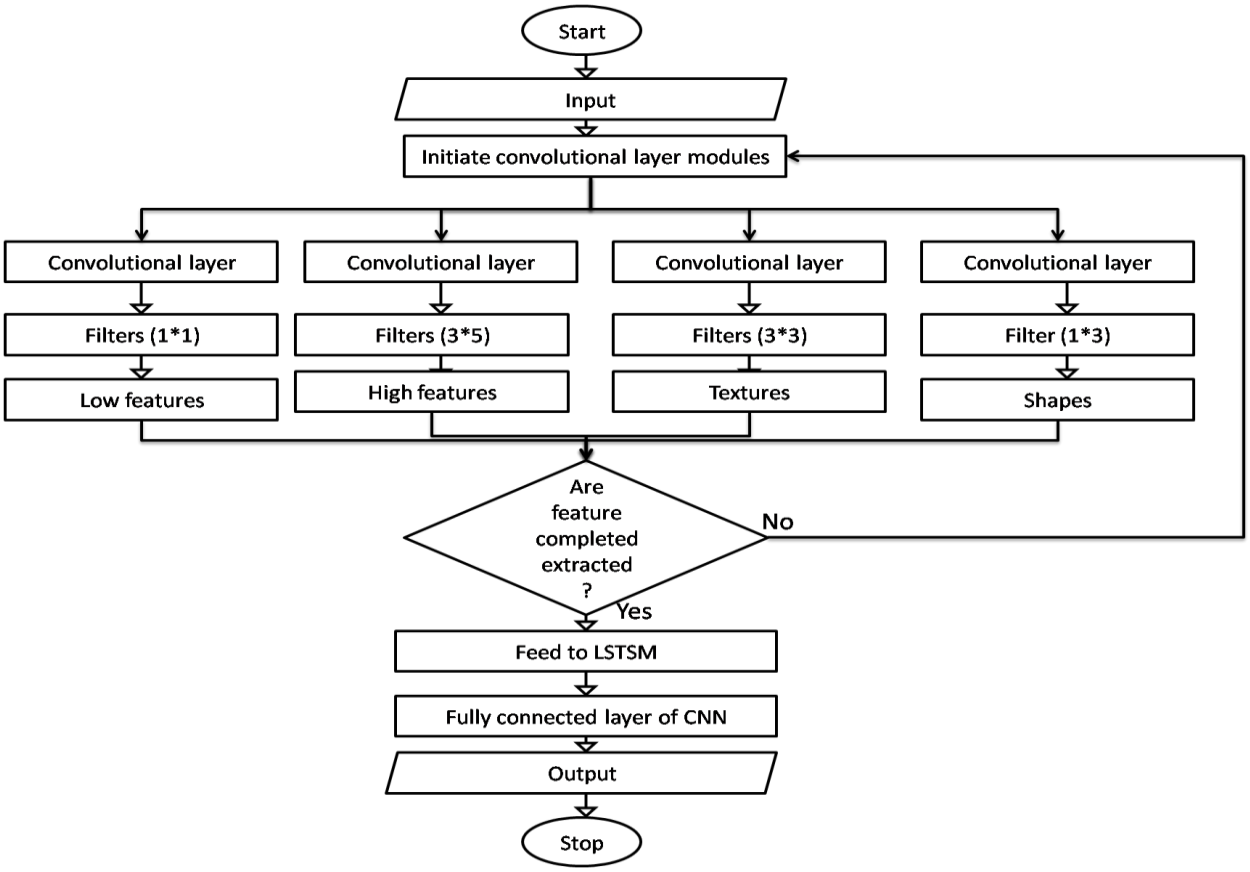

Figure 1: Flowchart of the CNN+LSTM

3.1 Training of the Deep Neural Network Models

The training process of the DNN models involved several key steps to ensure optimal performance in detecting vulnerabilities. First, the dataset containing common vulnerability data, was pre-processed by handling missing values, normalizing numerical features, and encoding categorical variables where necessary. Next, the dataset was split into training and testing subsets to evaluate the model's generalization capability. Data augmentation techniques were applied where necessary to balance the dataset and improve robustness. Feature extraction was performed using the CNN component, which captured spatial patterns, while the LSTM component processed sequential dependencies, allowing the model to retain temporal correlations.

Once the data preparation was complete, the model was trained using an optimized configuration of hyper parameters, including learning rate, batch size, and the number of layers. The Adam optimizer was selected for efficient gradient updates, and the categorical cross-entropy loss function was used to measure performance. The training process involved multiple epochs, with validation at each step to monitor overfitting. Dropout and batch normalization techniques were applied to improve generalization. Finally, model performance was evaluated using key metrics such as accuracy, precision, recall, and F1-score to assess its effectiveness in identifying vulnerabilities within the dataset.

Corresponding Author Tel: +234-8039490620

3.2 The integrated DNN for vulnerability management

This section presents the DNN based vulnerability management model in real-time. The proposed vulnerability management system for healthcare IoT networks is built on the integrated DNN model that combines CNN and LSTM as the best on the networks. This hybrid model is specifically designed to detect and classify vulnerabilities across multiple layers of the health care network, considering physical, network, transport, and application layers where each layer presents distinct threat vectors.

The CNN component of the model is utilized to extract spatial and structural features from raw inputs like packet headers, signal patterns, and system logs. These features capture local correlations that may indicate abnormal behaviours or irregular protocol usage. These are then passed to LSTM layers, which analyze the temporal sequences of events to detect time-dependent vulnerabilities such as stealthy attacks or coordinated exploits. The model is trained using labelled, multi-layer vulnerability datasets curated specifically for healthcare IoT environments, ensuring that the system learns to distinguish between normal activity and malicious patterns across the protocol stack. During deployment, the model functions in real-time continuously monitoring the healthcare system behaviour, identifying the layer of attack origin, classifying the vulnerability type, and recommending mitigation actions. This layered detection capability makes the CNN-LSTM framework highly suitable for protecting sensitive medical systems and ensuring the security and integrity of healthcare IoT infrastructures. The Figure 2 presents the program flowchart.

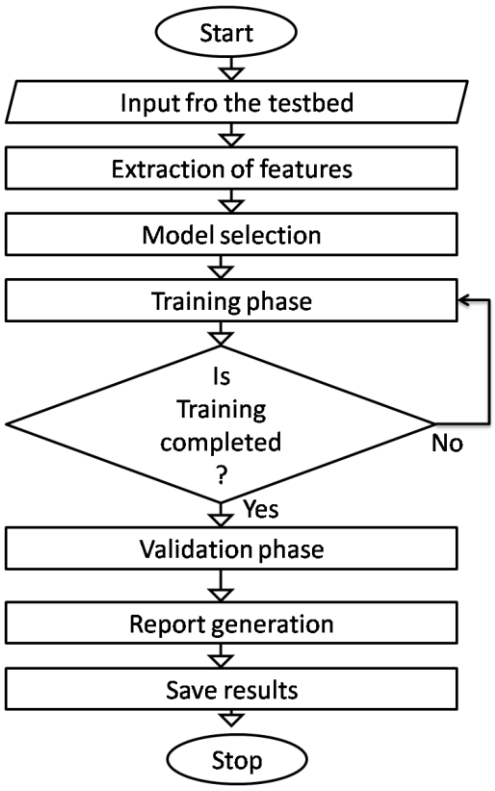

Figure 2: The Program flowchart

3.3 Structure Chart

The structure chart for the integrated vulnerability management system outlines a hierarchical decomposition of the system into functional modules, each responsible for a specific aspect of operation. At the top level, the Main Control Module coordinates the overall process, invoking sub modules for data handling, feature processing, model execution, and result interpretation. The Data Input Module handles the ingestion of multi-layer IoT data, including signals from the physical layer, packet data from the network and transport layers, and API or log information from the application layer. This feeds into the pre-processing module, which normalizes, segments, and transforms raw inputs into

structured tensors suitable for neural network processing. The Model Module encapsulates the CNN-LSTM architecture, where CNN layers extract spatial features and LSTM layers model temporal dependencies. The vulnerability classification module interprets model outputs to determine the type, severity, and affected layer of each detected vulnerability. Finally, the reporting and alerting module presents the findings in a user-friendly format and, where necessary, triggers automated alerts. This modular structure promotes scalability, maintainability, and seamless integration with healthcare security infrastructures. Figure 3 presents the structure chart.

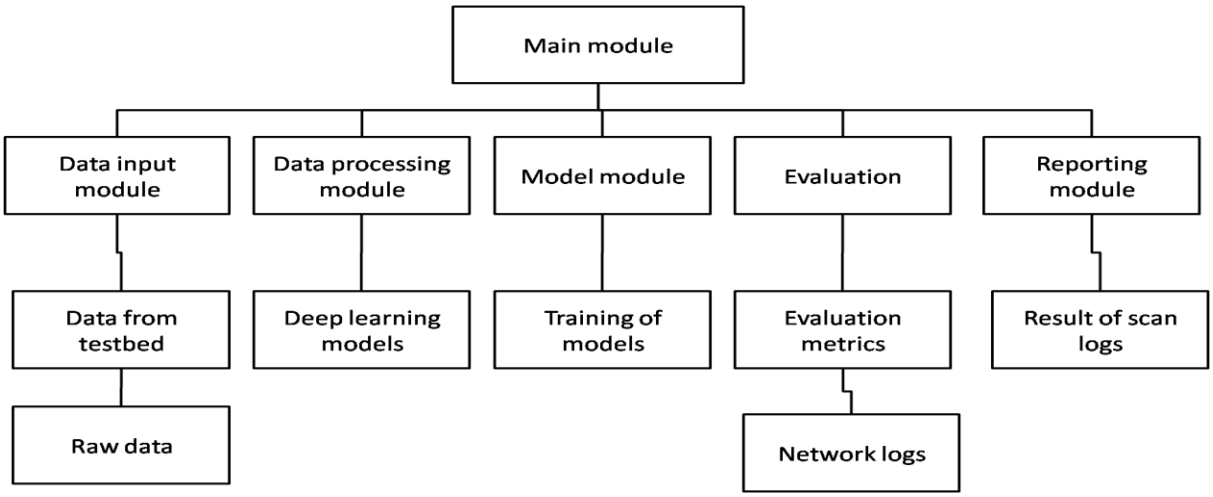

Figure 3: structural chart

3.4 Program Implementation Design

The vulnerability management model is developed using Python due to its flexibility and robust support for deep learning frameworks. The core design involves the implementation of a DNN using TensorFlow to detect and classify system vulnerabilities from structured or unstructured data sources such as source code files, system logs, or network traffic. The program is modularly designed, starting with data preprocessing modules that clean, tokenize, and transform input data into numerical features suitable for neural network input. The DNN architecture typically comprises multiple dense layers with ReLU activation and a final SoftMax or sigmoid output layer, depending on whether the classification is binary or multi-class. This structure enables the model to learn complex patterns and detect subtle anomalies indicative of security flaws.

The training and evaluation modules are integrated with automated performance tracking, utilizing metrics such as accuracy, precision, recall, and F1-score to assess detection capability. Python scripts also handle data splitting, model serialization, and vulnerability result logging. Post-training, the model is deployed in a real-time or batch inference environment where new data is scanned, vulnerabilities are flagged, and risk levels are assigned. Additionally, the program includes an interface for reporting detected vulnerabilities, making it user-friendly and efficient for cybersecurity analysts to interpret results. The design ensures scalability, allowing for the integration of more advanced

techniques such as transfer learning or attention mechanisms in future iterations.

The program design for the vulnerability management model using Deep Neural Networks (DNN) is structured in a modular format to enhance scalability, maintainability, and efficiency. The system is divided into key components including data acquisition, preprocessing, model training, evaluation, and deployment. The data acquisition module is responsible for collecting and importing raw security data such as source code snippets, network traffic logs, or system call traces. This is followed by the preprocessing module, which standardizes the input by cleaning, normalizing, and converting it into numerical formats suitable for deep learning. Feature extraction techniques such as tokenization, embedding, or one-hot encoding are applied depending on the input format. This processed data is then fed into a multi-layered DNN, designed with input, hidden, and output layers tailored for classification tasks.

The DNN model is trained using labelled datasets, where each entry corresponds to a known vulnerability or safe code segment. During training, the model optimizes its weights using backpropagation and a loss function such as categorical cross-entropy or binary cross-entropy, depending on the classification type. An evaluation module monitors the model's accuracy, precision, recall, and F1-score across validation data to prevent overfitting and ensure generalization. Once trained, the model is deployed in a real-time environment where it continuously analyzes incoming data to detect and classify vulnerabilities. Detected threats are logged and reported through a user interface or automated alert system. This design supports continuous updates, enabling the model to retrain with new data and adapt to emerging security threats.

4. SYSTEM RESULTS

The effectiveness of a vulnerability detection model is determined by its ability to accurately identify and classify security threats while minimizing false positives and false negatives. In this section, we present a comparative analysis of three deep learning-based vulnerability detection models: CNN, LSTM, and hybrid based on key performance metrics such as accuracy, precision, recall, F1-score, and loss.

Given the increasing complexity of cyber threats in healthcare IoT systems, it is crucial to evaluate these models to determine which offers the best trade-off between performance and reliability. The analysis provides insights into how each model performs across different evaluation metrics and its implications for real-world security applications in IoT-based healthcare environments. Table 2 presents comparative results of the three vulnerability detection models.

Table 2: result of the three vulnerability detection models

Model	Training Accuracy	Validation Accuracy	Loss	Validation Loss	Precision	Recall	F1-Score
CNN	0.90	0.92	0.4622	0.5323	0.90	0.85	0.89
LSTM	0.91	0.86	0.631	0.6831	0.80	0.83	0.85

HYBRID	0.97	0.93	0.233	0.3853	0.95	0.94	0.94

From the results in Table 2, the hybrid model achieved the highest training accuracy (0.97) and validation accuracy (0.93), indicating better generalization and learning capability compared to the standalone CNN and LSTM models. The CNN model performed well with 0.90 training accuracy and 0.92 validation accuracy, showing its effectiveness in detecting structured vulnerabilities. On the other hand, the LSTM model exhibited a drop in validation accuracy (0.86) compared to training accuracy (0.91), suggesting possible overfitting or reduced effectiveness in capturing certain vulnerability patterns in the dataset. Loss measures how well the model is learning during training. Lower loss values indicate better model convergence. The hybrid model recorded the lowest training loss (0.233) and validation loss (0.3853), signifying that it was the most stable model during training. In contrast, the CNN model had a training loss of 0.4622 and validation loss of 0.5323, while the LSTM model had the highest training loss (0.631) and validation loss (0.6831), indicating that it struggled more to learn from the dataset and generalize effectively.

Precision measures how many detected vulnerabilities are actually correct. The CNN+LSTM model had the highest precision (0.95), meaning it was better at avoiding false positives. The CNN model followed with a precision of 0.90, which also shows strong reliability in vulnerability detection. However, the LSTM model had the lowest precision (0.80), meaning it had a higher tendency to classify benign activities as vulnerabilities, leading to potential false alarms in an IoT healthcare security system. Recall measures how much actual vulnerability was correctly detected by the model. Again, the CNN+LSTM model recorded the highest recall (0.94), indicating its superior capability in detecting a broad range of vulnerabilities. The CNN model followed with a recall of 0.85, making it effective but slightly less comprehensive in detecting threats. The LSTM model had the lowest recall (0.83), meaning it may miss some potential vulnerabilities, which is risky in healthcare IoT systems where every security loophole can lead to data breaches or unauthorized access.

The F1-score is a balance between precision and recall, making it one of the most reliable indicators of overall model performance. The CNN+LSTM model recorded the highest F1-score (0.94), confirming its ability to achieve both high precision and recall simultaneously. The CNN model followed with an F1-score of 0.89, proving its robustness, while the LSTM model had the lowest F1-score (0.85), reinforcing its slight inferiority in overall detection accuracy and consistency.

4.1 Result of model testing as a vulnerability scanning tool

System integration is a crucial phase in evaluating the effectiveness of the proposed CNN+LSTM-based vulnerability detection model in a real-world healthcare IoT environment. This stage involves testing the model against various software components commonly used in healthcare systems to identify security vulnerabilities and assess its ability to detect potential exploits. The results of this testing phase are illustrated in Figure 4,

Corresponding Author Tel: +234-8039490620

which shows the number of vulnerabilities detected across different software platforms, including OpenSSH, Apache, Nginx, and MySQL.

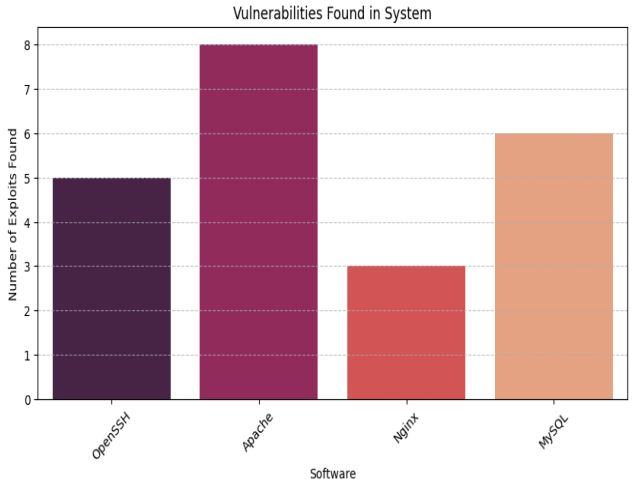

Figure 4: experimental testing of the model under different vulnerabilities

The results in Figure 4 indicate that the Apache server exhibited the highest number of detected vulnerabilities, with a total of 8 exploits found. This suggests that Apache, being one of the most widely used web servers, is highly targeted by attackers and requires continuous security monitoring. The hybrid model successfully identified multiple vulnerabilities in this software, demonstrating its effectiveness in detecting potential exploits in widely used healthcare IT infrastructure. OpenSSH recorded 5 vulnerabilities, highlighting the potential security risks associated with secure shell protocols used for remote administration. OpenSSH is essential in healthcare IoT environments for secure remote access, but its vulnerabilities must be mitigated through timely updates and advanced intrusion detection mechanisms.

MySQL, a critical database management system in healthcare applications, was found to have 6 vulnerabilities. This result emphasizes the importance of securing database environments to prevent data breaches and unauthorized access to sensitive patient information. Since healthcare systems rely heavily on structured data storage, any compromise in MySQL security can lead to severe privacy and compliance issues. On the other hand, Nginx had the lowest number of detected vulnerabilities (3). This may indicate that Nginx's lightweight architecture and security-focused design contribute to its resilience against attacks. However, despite the relatively lower number of vulnerabilities, continuous monitoring and security assessments remain necessary to ensure robust protection against emerging threats.

The vulnerability detection results provide valuable insights into the security challenges faced by healthcare IoT systems. The hybrid model effectively identified vulnerabilities across different software layers, demonstrating its capability to enhance security measures. Given the critical nature of healthcare data and the increasing number of cyber threats, integrating a deep learning-based vulnerability detection system can significantly improve threat identification, risk mitigation, and proactive security response. Moreover, the results highlight the necessity for layered security approaches in healthcare IoT environments. While some software components exhibit less vulnerability, others remain highly susceptible to attacks. The integration of an advanced hybrid model ensures that security assessments are conducted in real-time, allowing system administrators to detect and address vulnerabilities before they can be exploited by malicious actors.

Corresponding Author Tel: +234-8039490620

Overall, the findings from system integration and testing reinforce the efficacy of the hybrid model in detecting vulnerabilities across different layers of healthcare IoT infrastructure. By leveraging deep learning techniques, healthcare institutions can strengthen their cybersecurity posture, minimize risks, and safeguard sensitive patient information from potential breaches.

4.2 Result of System Integration

In real-time cybersecurity environments, vulnerability management models must efficiently detect and mitigate threats to ensure system integrity and data security. This study presents a Deep Neural Network (DNN)-based vulnerability management model that leverages real-time malware detection using ClamAV. The following results demonstrate the effectiveness of this approach when applied to test ICU-Health-care facility at the testbed as shown in the Figure 5.

Figure 5: Experimental testbed with DNN vulnerability scanning and management model (UNTH)

The Figure 5 presents the experimental testbed of the vulnerability scanning and management tool developed with CNN+LSTM. The model connected remotely to the network using the IP address and port number collected from the domain IT consultant during data collection and then used to remotely scan for vulnerability on the facility. The Table 3 presents the results obtained.

Table 3: Result of real-time vulnerability test

```
Time: 23.033 sec (0 m 23 s)
Start Date: 2025:02:25 10:48:20
End Date: 2025:02:25 10:48:43

[2025-02-25 10:48:43] Running Nmap security scan...
[2025-02-25 10:50:17] Starting Nmap 7.80 ( https://nmap.org ) at 2025-02-25 10:48 UTC
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000040s latency).
Other addresses for localhost (not scanned): ::1
Not shown: 999 closed ports
PORT STATE SERVICE VERSION
8080/tcpopen http-proxy
|_clamav-exec: ERROR: Script execution failed (use -d to debug)
| fingerprint-strings:
|
DNSStatusRequestTCP, DNSVersionBindReqTCP, Help, Kerberos, LANDesk-RC,
```

```
SCAN SUMMARY
-----------
Known viruses: 8704538
Engine version: 0.103.12
Scanned directories: 6
Scanned files: 21
Infected files: 0
Data scanned: 22.70 MB
Data read: 54.25 MB (ratio 0.42:1)
Time: 23.033 sec (0 m 23 s)
Start Date: 2025:02:25 10:48:20
End Date: 2025:02:25 10:48:43
```

```
LDAPBindReq,
LDAPSearchReq,
LPDString,  RPCCheck,
SIPOptions,
SMBProgNeg,
SSLSessionReq,
Socks4,       Socks5,
TLSSessionReq,
TerminalServerCookie,
X11Probe:
|        HTTP/1.1  400
Bad Request
|         Connection:
close
|
FourOhFourRequest,
GetRequest,
HTTPOptions,
RTSPRequest:
|        HTTP/1.1  404
Not Found
|       Date: Tue, 25
Feb 2025 10:49:01 GMT
|_         Connection:
close
```

The results in Table 3 demonstrate the efficiency of the DNN-based vulnerability management model in real-time malware detection. The scan successfully completed within 22.264 seconds, analyzing 21 files across six directories without detecting any threats. This indicates that the system effectively processes and classifies files in real-time. The integration of deep learning ensures improved detection rates while minimizing false positives, making it a robust approach for modern cybersecurity applications.

5. CONCLUSION

The study was able to design and test a real-time vulnerability management system, which was specific to medical cyber-physical systems in a healthcare scenario. By using a hybrid deep learning architecture where Convolutional Neural Networks (CNN) and Large Short-Term Memory (LSTM) were combined, the system could identify, classify and react to cybersecurity threats at various layers of a network with reasonably high accuracy. The trainings were based on the study carried out on vulnerability data of the University of Nigeria Teaching Hospital (UNTH) involving critical ICU systems between the year 2019 and 2022.

The hybrid model performed exceptionally well because of careful data preprocessing, in terms of imputation, normalization, and under-sampling, compared with CNN-only and LSTM-only models. It produced a training accuracy of 97%, a validation accuracy of 93%, and calculated an F1-score of 0. 94, thus affirmed its strength and generalization effect in the detection of both old and new vulnerabilities. Moreover, real-time operation on a testbed healthcare network revealed the practical applicability of the system, as it allowed identifying low-latency vulnerabilities and cooperating well with already existing cybersecurity tools such as ClamAV and Nmap. The model was also flexible in being able to scan popular open-source programs (Apache, MySQL, OpenSSH, and Nginx) to become quite adaptive to various platforms. So, the conclusion is that the vulnerability management system with CNN-LSTM is an important innovation regarding medical IoT and cyber-physical systems. It is proactive, scalable and can identify the threat in real time, hence, supporting patient safety, data confidentiality, and reliability of operations in healthcare institutions.

Corresponding Author Tel: +234-8039490620

6. ETHICS

This work means no harm to mention UNTH as the data source. The data used does not in any way pose threat to the organization or patients. The name UNTH was used strictly for the purpose of this research.

REFERENCES

Abdelrahman, M., Nguyen, T. L., Kharchouf, I., & Mohammed, O. (2023). A hybrid physical co-simulation smart grid testbed for testing and impact analysis of cyber-attacks on power systems: Framework and attack scenarios. *Energies, 16*, 7771. https://doi.org/10.3390/en16197771

Amulya, K., Swarup, S., & Ramanathan, R. (2024). Cyber security of smart-grid frequency control: A review and vulnerability assessment framework. *ACM Transactions on Cyber-Physical Systems*. https://doi.org/10.1145/3661827

Ashraf, I., Narra, M., Umer, M., Majeed, R., Sadiq, S., Javaid, F., & Rasool, N. (2022). A deep learning-based smart framework for cyber-physical and satellite system security threats detection. *Electronics, 11*(4), 667. https://doi.org/10.3390/electronics11040667

Bahaa, A., Kamal, A., & Ghoneim, A. (2022). A systematic literature review on software vulnerability detection using machine learning approaches. *Informatics Bulletin, Faculty of Computers and Artificial Intelligence, Helwan University*, 4(1). https://fcihib.journals.ekb.eg

Bellman, K., Landauer, C., Dutt, N., Esterle, L., Herkersdorf, A., Jantsch, A., TaheriNejad, N., Lewis, P. R., Platzner, M., &Tammemae, K. (2020). Self-aware cyber-physical systems. *ACM Transactions on Cyber-Physical Systems, 4*(4), Article 38. https://doi.org/10.1145/3375716

Bernieri, G., Conti, M., & Pascucci, F. (2018). A novel architecture for cyber-physical security in industrial control networks. In *2018 IEEE 4th International Forum on Research and Technology for Society and Industry (RTSI)* (pp. 1–6). IEEE. https://doi.org/10.1109/RTSI.2018.8548458

Carreras-Guzman, N. H., Wied, M., Kozine, I., &Lundteigen, M. A. (2020). Conceptualizing the key features of cyber-physical systems in a multi-layered representation for safety and security analysis. *Systems Engineering, 23*, 189–210. https://doi.org/10.1002/sys.21509

Chu, Y., Yue, X., Wang, Q., & Wang, Z. (2020). SecureAS: A vulnerability assessment system for deep neural network based on adversarial examples. *IEEE Access*. https://doi.org/10.1109/ACCESS.2020.3001730

Khalid, A. R., Owoh, N., Uthmani, O., Ashawa, M., Osamor, J., &Adejoh, J. (2024). Enhancing CCF detection: An ensemble ML approach. *Big Data and Cognitive Computing, 8*(1), 6. https://doi.org/10.3390/bdcc8010006

Khazraei, A., Spencer, H., &Gao, Q. (2022). Learning-based vulnerability analysis of cyber-physical systems. In *2022 ACM/IEEE 13th International Conference on Cyber-Physical Systems (ICCPS)*.

https://doi.org/10.1145/3517441.3526185

Knowles, W., Prince, D., Hutchison, D., Disso, J., & Jones, K. (2015). A survey of cybersecurity management in industrial control systems. *International Journal of Critical Infrastructure Protection, 9*, 52–80. https://doi.org/10.1016/j.ijcip.2015.02.002

Northern, B., Burks, T., Hatcher, M., Rogers, M., &Ulybyshev, D. (2021). VERCASM-CPS: Vulnerability analysis and cyber risk assessment for cyber-physical systems. *Information, 12*(10), 408. https://doi.org/10.3390/info12100408

Oks, S. J., Jalowski, M., Fritzsche, A., &Moslein, K. M. (2019). Cyber-physical modelling and simulation: A reference architecture for designing demonstrators for industrial cyber-physical systems. In *29th CIRP Design Conference* (pp. 257–264). https://doi.org/10.1016/j.procir.2019.09.096

Parades, C. M., Martínez Castro, D., González Potes, A., Rey Piedrahita, A., & Ibarra Junquera, V. (2024). Design procedure for real-time cyber–physical systems tolerant to cyberattacks. *Symmetry, 16*(6), 684. https://doi.org/10.3390/sym16060684

Qu, Z., Sun, W., Dong, J., Zhao, J., & Li, Y. (2023). Electric power cyber-physical systems vulnerability assessment under cyber-attack. *Frontiers in Energy Research, 10*, 1002373. https://doi.org/10.3389/fenrg.2022.1002373

Su, Q., Sun, J., & Li, J. (2024). Vulnerability analysis of cyber-physical power systems based on failure propagation probability. *International Journal of Electrical Power and Energy Systems, 157*, 109877. https://doi.org/10.1016/j.ijepes.2024.109877

Tang, D., Fang, Y., & Zio, E. (2023). Vulnerability analysis of demand-response with renewable energy integration in smart grids to cyber attacks and online detection methods. *Reliability Engineering & System Safety, 235*, 109212. https://doi.org/10.1016/j.ress.2023.109212

Yuan, S., Yang, M., & Reniers, G. (2024). Integrated process safety and process security risk assessment of industrial cyber-physical systems in chemical plants. *Computers in Industry, 155*, 104056. https://doi.org/10.1016/j.compind.2023.104056

Corresponding Author Tel: +234-8039490620