Journal on
Blockchain Intelligent Computing

Tech Science Press

https://doi.org/10.32604/jbic.2026.081492

ARTICLE

A Blockchain-Based Adaptive Security Framework with Real-Time Incident Response and Usability Feedback for Non-Expert Users

Mosammat Jannatul Kobra[1], **Muhammad Rashid Majeed**[2,*] **and Md Owahedur Rahman**[1]

[1]School of Electronics and Information Engineering, Nanjing University of Information Science & Technology, Nanjing, China
[2]School of Computer Science, Nanjing University of Information Science & Technology, Nanjing, China

*Corresponding Author: Muhammad Rashid Majeed. Email: rashid-majeed@outlook.com

Received: 04 March 2026; Accepted: 20 April 2026; Published: 13 May 2026

ABSTRACT: The proposed study introduces a blockchain-based framework for an adaptive security solution with real-time incident response and usability feedback for non-expert users. Traditional security solutions are often designed with static, opaque policies, which makes them complex. These issues make them less effective in dealing with complex environments. Thus, to make them more effective, the proposed framework introduces supervised machine learning for attack classification, unsupervised machine learning for anomaly detection, a risk-aware, adaptive policy engine, and a lightweight, tamper-evident, hash-linked ledger for auditable decision-making. The proposed framework uses a Random Forest classifier for BENIGN/ATTACK classification, and an Isolation Forest module for anomaly detection. The proposed framework also uses an adaptive policy engine to decide whether to use HIGH or LOW security mode based on a combined risk score calculated from the probability of an attack and the rate of anomalies. Moreover, the proposed framework incorporates usability feedback through interaction steps, user errors, and setup time, making it suitable for non-expert users. The proposed framework was implemented using the CICIDS2017 benchmark dataset for intrusion detection, which contains 2,830,743 flows with 70 numerical features from 8 CSV files. The proposed framework achieved an accuracy of 0.9789, precision of 0.9753, recall of 0.9793, F1-score of 0.9773, and ROC-AUC of 0.9799 using the Random Forest classifier. The proposed framework achieved an anomaly rate of 0.1456 using the Isolation Forest module and a risk score of 0.1824 using the adaptive policy engine, which decides whether to use LOW or HIGH security mode. Moreover, the proposed framework enabled a simple interaction process, making it suitable for non-expert users.

KEYWORDS: Blockchain-based security; adaptive security framework; real-time anomaly detection; usability-aware security; machine learning-driven policy adjustment

1 Introduction

The increasing use of data-intensive and intelligent systems has substantially increased the complexity of security management. Machine learning-based and autonomous decision-making systems are increasingly being used in various fields such as healthcare, finance, smart infrastructure, and cyber-physical systems. Although these systems have enhanced efficiency and prediction accuracy, they have also introduced new security risks, such as data manipulation, model attacks, and delayed responses to security incidents [1,2]. Traditional security systems typically rely on static security policies, predefined thresholds, and manual security auditing. However, these systems are not effective in dynamic environments where the system's behavior, data patterns, and attack dynamics are constantly changing [3,4]. In addition, static security

Copyright © 2026 The Authors. Published by Tech Science Press. This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

systems may not provide the level of decentralization and peer-to-peer verification found in blockchain-based systems, making it difficult to conduct forensic analysis in the event of a security attack. Therefore, there is a need to develop adaptive security systems [5].

Machine learning-based security systems have been proposed to enhance blockchain platforms by providing real-time automated threat detection and predictive risk analysis [6]. For example, anomaly detection techniques such as Isolation Forest and one-class classification have shown promising results in detecting unusual system behavior without requiring attack data [7]. However, the majority of existing ML-based security systems have focused primarily on accuracy, neglecting two important factors: auditability and usability. The absence of a proper audit trail makes it impossible to verify security decisions independently. On the other hand, a lack of usability awareness may make it difficult for non-expert users to interact with the security system, leading them to distrust it altogether [8]. Recently, blockchain technology has been proposed as a solution for addressing auditability and integrity concerns in distributed systems. Due to its inherent characteristics of immutability and tamper-resistance, blockchain technology has been increasingly used for security logging, access control, and decision traceability. In this regard, security-related events and decisions can be recorded on a blockchain ledger. However, existing blockchain-based security solutions have been isolated from ML models and have not been integrated with adaptive security policies [9]. Another important yet relatively unexplored issue in security system design is usability for non-expert users. The majority of security frameworks have been designed for technically skilled users [10]. Previous research has observed that overly stringent security constraints may reduce user efficiency and increase operational errors, especially in scenarios that involve frequent human intervention [11]. Therefore, there is an emerging need for security solutions that can effectively strike a balance between protection efficacy and usability.

To address these problems, the research proposes a novel Blockchain-Based Adaptive Security Framework with Real-Time Incident Response and User Feedback. The proposed approach is inspired by optimization algorithms used to address the complexities of solving major scientific problems and comprises a supervised learning algorithm for attack classification, unsupervised learning for flow-based anomaly detection, and a policy-driven framework for real-time incident response that adapts to dynamic changes in security strength [12]. In this proposed solution, a Random Forest classifier is used to differentiate benign and malicious traffic patterns. An Isolation Forest module is used to detect anomalous traffic patterns. Every security-related action, including model performance metrics, identified anomalies, policy decisions, and response actions, is recorded in a lightweight, tamper-evident, hash-linked ledger that follows blockchain-oriented design principles [13]. Unlike existing approaches, the suggested framework takes into consideration the evaluation criteria related to the usability of the system, like the steps users take to interact with the system, errors, and setup time, in the decision-making process related to security. Feedback on these criteria is used to relax security policies, if needed, to avoid oversecurity and to maintain continuous monitoring. The experimental evaluation has been conducted by utilizing the CICIDS2017 benchmark intrusion detection dataset [14]. The findings related to the suggested framework indicate that it is useful for attack classification, anomaly awareness, decision logging, and usability awareness in cybersecurity.

The contributions of this paper can be summarized in three ways:

- A cybersecurity-focused adaptive security framework that supports supervised attack classification, anomaly detection, incident response policy selection, and usability-informed feedback.
- A risk-based policy engine that utilizes predicted probabilities of attacks and anomaly rates to trigger HIGH or LOW security modes rather than relying on regression accuracy.
- A tamper-evident audit logging system to capture security decisions, anomaly detection results, and policy actions.

2 Literature Review

Significant improvements in the development of smart security systems have led to a substantial increase in the application of machine learning and blockchain-oriented technology in cybersecurity. Specifically, blockchain technology has been recognized as a promising technology in enhancing trust in distributed environments. The potential of blockchain technology to enhance trust in IoT systems, especially in environments where secure logging is a critical requirement, has been identified [15]. Subsequent research has further explored the security, privacy, and architecture of blockchain technology, which has been recognized as a promising technology for providing secure logging services across a range of application environments [16]. Further research-oriented studies on the implementation side have also demonstrated that permissioned blockchain platforms such as Hyperledger Fabric can facilitate structured, auditable transaction support in enterprise and security-critical environments [17]. However, recent studies on scalable blockchain consensus have also underscored the importance of accounting for performance and scalability challenges when implementing blockchain in real-world environments [18]. Collectively, these studies imply that blockchain-based designs have significant potential to facilitate security auditing, though their integration with adaptive machine-learning-based policy control remains limited.

Parallel to this, anomaly detection has emerged as a key research area in cybersecurity, especially in settings where attack patterns are dynamic and labeled malicious data may not always be available. In fact, some surveys on the broader field of anomaly detection have highlighted the effectiveness of detecting anomalies in complex settings by identifying deviations from normal behavior [19]. More recent research has highlighted the importance of explainability in anomaly detection, enabling outputs to be effectively utilized in decision-making scenarios [20]. The emergence of deep anomaly detection techniques, like Deep Isolation Forest, has enhanced the effectiveness of unsupervised anomaly detection in complex settings [21]. In the field of network intrusion detection, existing benchmarking studies using the CICIDS2017 dataset have demonstrated the potential of machine learning techniques for classification and anomaly detection [22]. This is particularly significant because it provides a more scientifically sound basis for evaluating intrusion detection systems than using non-security data. However, the majority of existing anomaly detection studies have focused on accuracy without sufficiently investigating How anomaly-related information should be mapped to adaptive policy enforcement, auditable security actions, and human-centric security management.

Usability has also been recognized as an essential factor of practical security system design. Research studies that have emphasized non-expert users have clearly demonstrated that awareness, perception, and participation in cybersecurity are significantly influenced by the usability of security mechanisms within real-world contexts. Systematic review studies that have emphasized the importance of usability have clearly demonstrated that it remains an essential issue that warrants greater attention in security design, especially when technology is used to support non-expert users. Research studies that have emphasized human-centric cybersecurity have clearly demonstrated that users are often treated as the weakest link when security mechanisms are designed without considering realistic behavior patterns, trust, and interaction constraints. Studies of compliance behavior at the organizational level have also shown that security effectiveness depends not only on the strength of security but also on users' ability to comply with security requirements [23]. In software design, lightweight, usable security has therefore been advocated as a principle for designing software systems that maintain security without imposing complexity on end users. Other studies on the design of secure systems have shown that communication between system designers and users is important for the adoption of security [24]. Other studies of security workers in AI and machine learning systems have emphasized the importance of designing security systems that support threat understanding, interpretation, and decision-making. In addition, the study on privacy and security by design supports the importance of integrating human and organizational aspects into the design process itself, rather than treating them as

secondary [25]. The primary usability tools, such as SUS and NASA-TLX, also appear more relevant in the current context, as they help evaluate usability, workload, and human effort in a system.

Research in machine learning-based cybersecurity has also continued to demonstrate the effectiveness of intelligent detection techniques in enhancing security in networked environments. Surveys on machine learning and deep learning in cybersecurity have shown the effectiveness of learning-based techniques for robust attack detection, traffic classification, and even automated threat analysis across a wide range of applications [26]. More recent research on network anomaly intrusion detection using deep learning has shown the effectiveness of learning-based techniques for strong discrimination between benign and attack traffic in real-world networks [27]. Similar trends have also been observed in other domain-specific scenarios, such as the Internet of Medical Things, where machine learning-based intrusion detection has been identified as a key security enabler [28]. Although these studies have shown promising detection accuracy, they have primarily focused on the predictive aspects of intrusion detection, without offering an integrated framework that includes attack classification, anomaly-based policy adaptation, tamper-evidence-based audit logging, and usability-based security controls.

Although considerable progress has been achieved in existing research on trust mechanisms, anomaly detection, intrusion detection, and security using blockchain technology [29], these components are still largely studied independently. Research on blockchain technology has focused on issues such as integrity, traceability, and decentralization, but has not been extensively explored for incorporating adaptive security policies that leverage real-time evidence of attacks [30]. Research on anomaly detection and intrusion detection systems has shown promising results in their technical capabilities [31], but they are largely evaluated based on their detection capabilities [32] rather than their contributions to auditable incident handling [33]. Meanwhile, existing usable security studies have demonstrated that overly complex or rigid security systems may lead to decreased user trust [34,35], increased error rates, and decreased compliance, especially among non-expert users [36]. However, these usability issues are rarely taken into account in closed-loop adaptive cybersecurity systems [37].

Thus, a research gap exists for developing security systems that integrate machine learning-based attack classification, anomaly-based risk estimation, adaptive control, decision-logging mechanisms, and usability. To address this research gap, this study proposes an adaptive security system that supports trustworthy, transparent, and user-friendly security decision-making for non-expert users in cybersecurity-relevant settings.

Research Gap

Even though various studies have proven the effectiveness of machine learning-based intrusion detection, anomaly detection, blockchain-based auditability, and usable security, these features are mostly examined individually. Existing solutions are mostly either strong in one area or weak in another. Machine learning-based solutions are strong in adaptability but weak in auditability; blockchain-based solutions are strong in auditability but weak in adaptability; and solutions related to usable security are mostly not integrated with automated threat response and detection [15,19,22]. In fact, recent anomaly-detection extension techniques, such as Deep Isolation Forest, demonstrate that ongoing developments in detection techniques are evident, yet their integration with audit-aware adaptive security policies remains limited [37]. This again shows the strong need for an integrated solution that incorporates all of these techniques into a single solution. This study is motivated by the need to propose a more integrated and realistic adaptive security solution for non-expert users.

Novelty of This Work

This study aims to bridge the identified research gap by proposing a unified, closed-loop concept of adaptive security that simultaneously incorporates decision evaluation via machine learning, real-time anomaly detection, blockchain-based immutable logging, and usability-informed security policy adaptation. Unlike other studies in this area, this proposed framework does not separate security, auditability, and usability into distinct, independent considerations; instead, all three are tightly integrated into a single framework. This is because the proposed framework can adapt security policies based on system behavior and user interaction metrics while maintaining a tamper-proof audit trail, thus pushing the boundaries of the existing art in this area.

3 Methodology

In our study, we propose a Blockchain-Based Adaptive Security Framework that incorporates real-time machine learning, adaptive security policy, and usability feedback for non-expert users. This framework will automatically detect security threats in real time.

Fig. 1 illustrates the overall workflow of the proposed blockchain-based adaptive security framework, from data acquisition to final security enforcement and persistence. The overall workflow begins with data source acquisition, where the CICIDS2017 benchmark intrusion detection dataset is acquired. The acquired data is then subjected to data preprocessing, including handling missing values and categorical encoding, followed by feature construction to generate the final input feature set. The data is then split into training and testing data to train the classification model. The classification model is then evaluated using classification performance metrics. The combined risk score is calculated based on the predicted attack probabilities and the anomaly rate. The decision to switch the security policy to HIGH or remain at LOW is made. Regardless of the policy level, the framework initiates a policy-aware automatic incident response module. If the security level is set to HIGH, proactive containment actions are initiated; when set to LOW, passive monitoring and logging are performed. Concurrently with the above, real-time anomaly detection is carried out, in which the received data is analyzed to identify anomalous behavior. The results of anomaly detection, model performance metrics, the applied security policy, and incident response actions are recorded in a lightweight, tamper-evident, hash-linked ledger. A new record is then added to the ledger to ensure the immutability and auditability of security-related decisions. Once tamper-evident hash-linked ledger logging is complete, a usability evaluation is conducted, during which metrics such as user steps, errors, and setup time are collected. A feedback check is conducted to determine whether user satisfaction falls below a specified threshold. If usability is not satisfactory, the security policy is adjusted to improve it; otherwise, the existing policy is retained. This completes the final security policy, which will control subsequent system behavior. Finally, the trained model and the tamper-evident hash-linked ledger are saved, and the process halts. This flowchart shows the integration of machine learning, anomaly detection, adaptive security policies, blockchain-based logging, and usability feedback in a closed-loop, user-centric security system.

3.1 Data Collection and Preprocessing

The experimental validation was performed using the CICIDS2017 intrusion detection benchmark dataset, which comprises 8 CSV files of network flows. During the preprocessing step, column names were standardized, numerical columns were converted as needed, incorrect values, such as positive and negative infinity, were replaced with missing data, empty columns were removed, constant columns were removed, and missing numerical values were replaced with the median. The target column was converted to binary format, with BENIGN mapped to 0 and ATTACK mapped to 1.

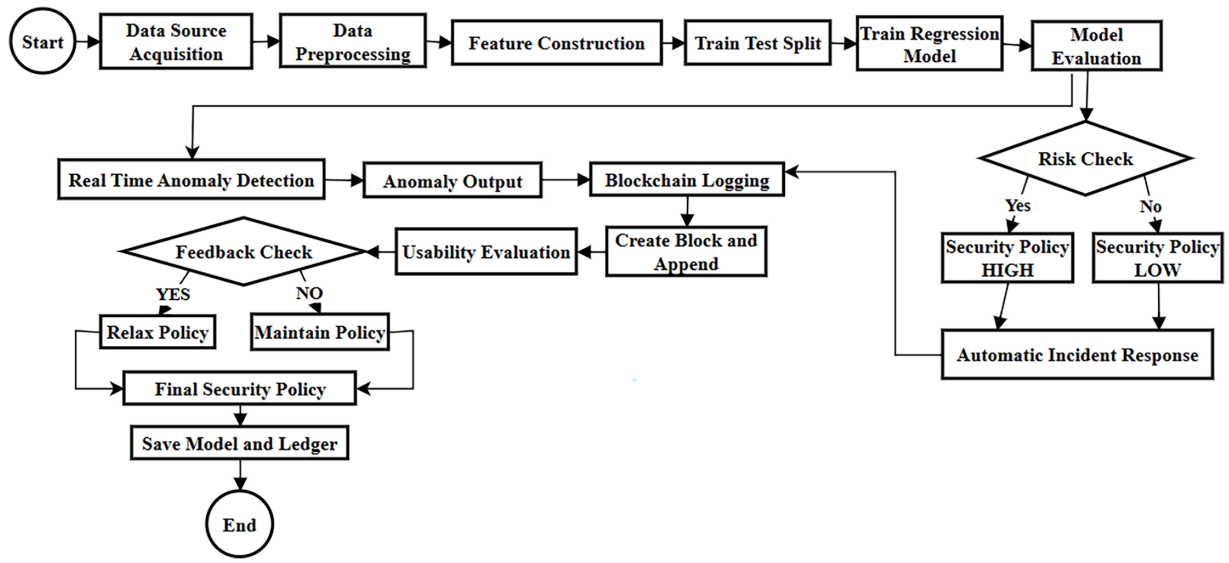

Figure 1: Flowchart of the proposed blockchain-based adaptive security framework with real-time incident response and usability feedback.

Fig. 2 shows a high-level diagram of the proposed blockchain-based adaptive security framework, which depicts the closed-loop process of model training, anomaly detection, security policy updates, tamper-evident hash-linked ledger logging, and usability feedback.

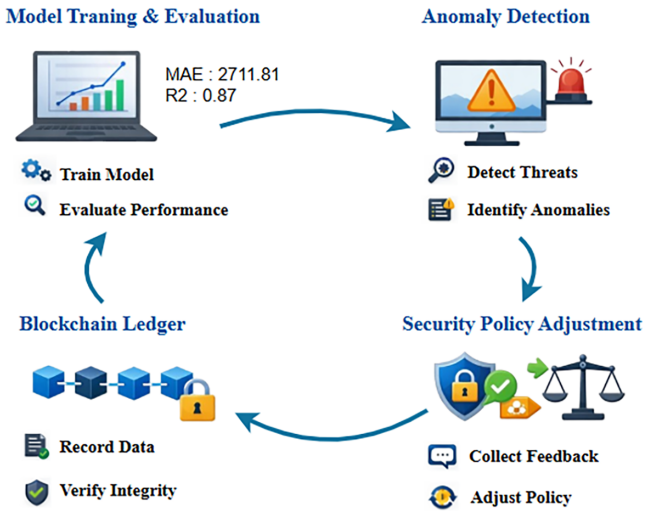

Figure 2: High-level overview of the proposed blockchain-based adaptive security framework with real-time anomaly detection and usability-driven policy adjustment.

Missing Value Handling

Missing numerical values are filled using the mean value:

$$x_i^{filled} = \begin{cases} x_i, & \text{if } x_i \neq \varnothing \\ \frac{1}{N}\sum_{j=1}^{N} x_j, & \text{otherwise} \end{cases} \tag{1}$$

This ensures no loss of training data due to missing entries.

Encoding Categorical Variables

Unlike the previously used data set, the CICIDS2017 data set contains mostly network flow attributes. Therefore, in this data set, attributes such as sex, smoker status, and region are not applicable. In the modified data preprocessing pipeline, first, all the feature names are normalized by removing extra spaces. Then, all flow attributes are converted to numeric values if possible. Next, invalid values, such as positive and negative infinity, are replaced with missing values. Then, empty and constant value rows are removed. Finally, missing values in numeric attributes are replaced with the attribute's median. In this data set, the target attribute is Label. It is converted into binary form with BENIGN = 0 and ATTACK = 1.

Table 1 presents the primary feature categories used in this research and their preprocessing techniques. The transformed data set contains network flow features derived from the CIC IDS 2017 benchmark data set for intrusion detection, rather than demographic and medical features.

Table 1: Feature descriptions and encoding methods.

Feature/Category	Type	Processing/Encoding
Network-flow attributes	Numerical	Converted to numeric format
Invalid values (Inf, −Inf)	Numerical anomalies	Replaced with missing values (NaN)
Missing values	Numerical	Median imputation
Empty columns	Non-informative	Removed
Constant columns	Non-informative	Removed
Label	Categorical target	Binary encoding (BENIGN = 0, ATTACK = 1)

3.2 Machine Learning Model for Binary Classification

The supervised learning part of the proposed framework is implemented using the Random Forest classifier for binary classification between BENIGN and ATTACK traffic. This is done using the preprocessed CICIDS2017 dataset, and the evaluation metrics are accuracy, precision, recall, F1-score, and ROC AUC. This classifier is the decision-making part of the proposed adaptive security framework.

The supervised learning part of the suggested framework utilizes a Random Forest classifier for binary classification of BENIGN and ATTACK traffic. The model is trained on the preprocessed CICIDS2017 dataset. Metrics used for model evaluation include accuracy, precision, recall, F1 Score, and ROC AUC. The trained model is used for decision-making in the suggested adaptive security model:

$$\hat{y} = \frac{1}{T}\sum_{t=1}^{T} f_t(x) \tag{2}$$

where T represents the number of trees and f_t is the prediction of the t- the tree.

Performance is evaluated using:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} | y_i - \hat{y}_i | \tag{3}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{4}$$

The trained model is then evaluated, and its performance metrics are recorded.

3.3 Real-Time Anomaly Detection

Anomaly detection is crucial for identifying unusual patterns in incoming data that could reveal security violations. We apply Isolation Forest, an unsupervised learning algorithm that isolates anomalies via random tree splits.

The anomaly scoring formula is:

$$s(x) = 2^{-\frac{E(h(x))}{c(n)}} \tag{5}$$

This formula is derived from the conventional Isolation Forest theory [35]. Data points with scores below a given threshold are labeled as anomalies. This module continuously observes input features in real time and produces a binary output indicating whether the activity is normal or anomalous.

3.4 Adaptive Security Policy Engine

The adaptive security policy engine uses a combined risk score derived from the predicted attack probability and the anomaly rate to determine the mode of operation. The combined risk score is a weighted combination of the predicted attack probability and the anomaly rate. If the combined risk score exceeds a threshold, the HIGH security mode is activated; otherwise, the LOW security mode is activated. This way, the decision is not based on any regression performance that may not be relevant to cybersecurity.

$$Policy = \begin{cases} HIGH, & R^2 < \tau \\ LOW, & R^2 \geq \tau \end{cases} \tag{6}$$

where τ is a predefined threshold for the combined risk score.

The policy is determined by a combined risk score, which is a function of the predicted attack probability and the anomaly rate, rather than the regression's performance. The HIGH security mode is activated by a higher risk score, while a lower risk score activates the LOW security mode.

Fig. 3 depicts the workflow of the proposed framework's risk-driven adaptive security. The framework first classifies and analyzes anomalies, providing the system with the predicted probability of an attack and the anomaly rate from the monitored network flow. The framework then uses the predicted attack probability and the anomaly rate to compute a risk score. The computed risk score is then compared with a predefined threshold. If the computed risk score exceeds the threshold, the framework activates the HIGH security policy, which in turn initiates active incident response. If the computed risk score is lower than the threshold, the framework activates the LOW security policy. The framework also logs security decisions, incident responses, and monitoring results in a lightweight, tamper-evident, hash-linked ledger.

3.5 Blockchain-Based Immutable Ledger

To ensure auditability and non-repudiation of security-related activities, all model evaluations, security decisions, and anomaly-detection outcomes are recorded in a lightweight, tamper-evident, hash-linked ledger.

Each new block contains:

- Dataset hash
- Model parameter hash
- Classification performance metrics

- Security policy
- Incident response actions
- Anomaly results

Block hashing is computed using:

$$Hash_i = SHA256(Block\_data_i) \tag{7}$$

The hash-linked structure ensures tamper evidence, accountability, and traceability of logged security events [37].

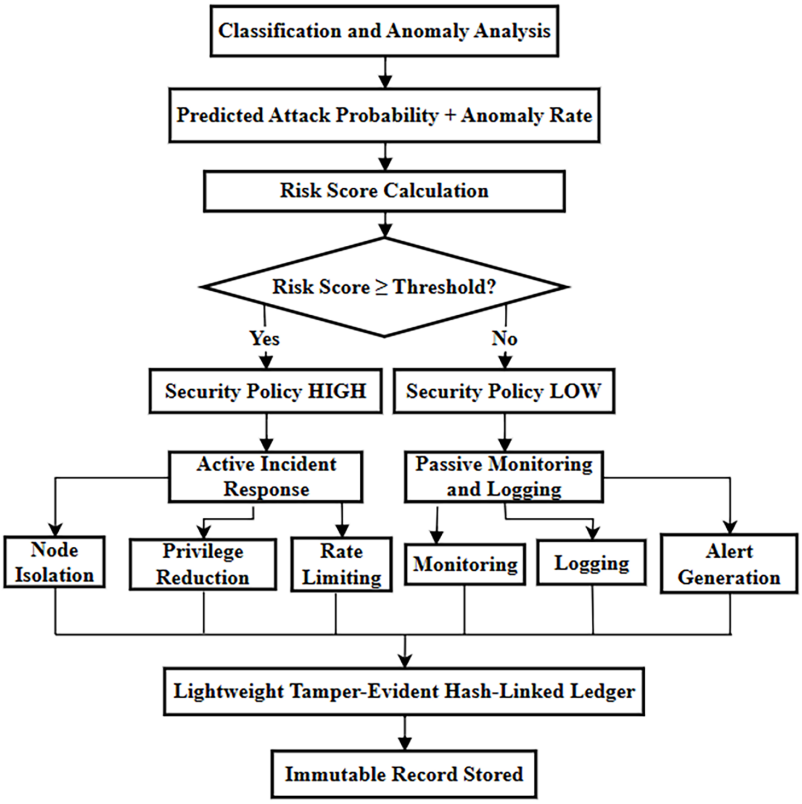

Figure 3: Policy-aware incident response sequence under HIGH and LOW security modes.

Fig. 4 shows the structure of the lightweight tamper-evident hash-linked ledger employed within the proposed framework. As shown, the proposed ledger will start with a Genesis Block that initializes the chain with a null value for the preceding hash. Each block will include the dataset hash value, the model hash value, performance metrics, risk score, anomaly rates, security policy, and incident response actions generated during the evaluation process. Moreover, each block will contain the hash of the preceding block, forming a hash-linked chain. This ensures the integrity, traceability, and immutability of security-related decisions and model outcomes over consecutive evaluation cycles.

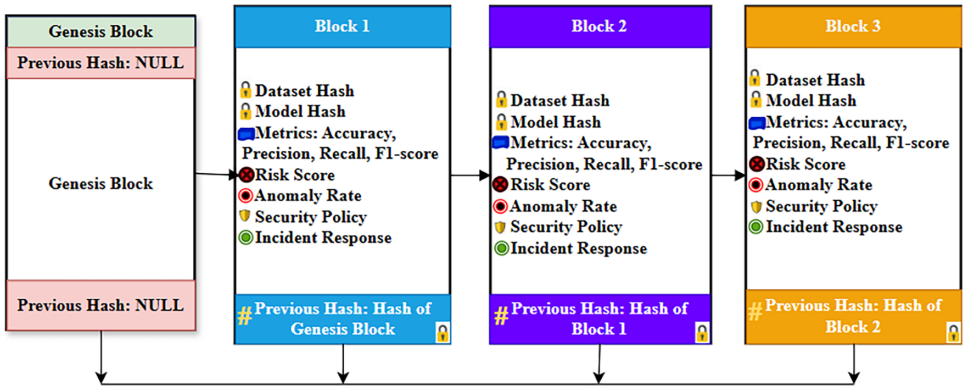

Figure 4: Policy-aware incident response sequence under HIGH and LOW security modes.

3.6 Preliminary Simulated Usability Assessment

Given that the target users of the proposed framework are non-experts, a preliminary simulated usability assessment has been included to determine the impact of interaction burdens on security policy decisions. Three operational measures were considered during the interaction workflow: the total number of user interaction steps, the number of user errors, and the setup time required to complete the process.

The reduced number of steps, user errors, and setup times were considered usability indicators. They were not considered validation measures; rather, they were considered supportive measures to determine whether the security policy to be applied should be strengthened or relaxed for non-expert usability.

The usability component of this framework, being a simulation-based component, produces outputs that are considered preliminary indicators only and not alternatives to standardized human-centered evaluation instruments. A more complete user study using standardized frameworks such as SUS and NASA-TLX is considered for future work.

User satisfaction score is defined as:

$$S = \max\left(1, \min\left(5, 5 - \left(E + 0.5 \cdot Steps\right)\right)\right) \tag{8}$$

Usability Indicators Used in the Preliminary Simulated Assessment.

The usability indicators used for the preliminary simulated evaluation are summarized as shown in Table 2. These usability indicators are intended to represent the simplicity of interaction and the operational burden for non-expert users, and are used only as supporting inputs for adaptive policy adjustment.

As shown in Table 3, each problem and solution will be represented within the framework's design, implementation, and evaluation. All these components will be part of adaptive security control, decision logging, and accessibility.

4 Results and Discussion

This section presents the experimental results obtained using the proposed blockchain-based adaptive security framework and analyzes them in terms of prediction accuracy, security dynamics, blockchain traceability, and usability for non-technical users.

Table 2: Usability metrics summary.

Indicator	Description	Interpretation
Steps	Number of interaction actions required to complete the workflow	Lower values indicate simpler interaction
Errors	Number of user mistakes during the interaction process	Lower values indicate better usability
Setup Time	Time required to complete the initial interaction and configuration process	Lower values indicate faster and easier use
Usability Signal	Internal control signal derived from the above indicators	Used only for preliminary policy adjustment

Table 3: Problems addressed and solutions.

Problem	Solution
Static security policies	Adaptive policy engine via ML & anomaly scores
Lack of real-time incident response	Automatic response (node isolation, privilege reduction)
Non-expert usability gap	Usability feedback loop supports simplified interactions
Security auditability	Lightweight tamper-evident hash-linked ledger records security events and decisions
Risk-aware security monitoring	Classification outputs and anomaly analysis support continuous risk estimation and policy selection

4.1 Model Performance Evaluation

The proposed framework is evaluated using the benchmark intrusion detection dataset from CICIDS2017. The dataset is first preprocessed and converted to a binary label. In the revised experiment design, the supervised learning component of the framework is realized using the Random Forest classifier to differentiate between BENIGN and ATTACK traffic classes. The problem formulation is changed from regression to classification; hence, the framework is evaluated using classification performance metrics, including accuracy, precision, recall, F1 score, and ROC AUC.

The experiment results show that the classifier achieves high predictive performance. The classifier achieves 0.9789 accuracy, 0.9753 precision, 0.9793 recall, 0.9773 F1 score, and 0.9799 ROC AUC values. These values show that the classifier can reliably differentiate between benign and malicious traffic patterns. The confusion matrix also demonstrates the classifier's robustness. Out of the test set, the classifier correctly classified 567,612 benign instances and 139,312 attack instances, producing only 662 false positives and 100 false negatives. This shows that the classifier maintains a low error rate while maintaining high sensitivity to malicious traffic.

The above results confirm the suitability of the proposed classifier as a reliable decision-making component of the suggested framework for adaptive security solutions.

The combined dataset contained 2,830,743 flow records with 70 numerical features after preprocessing.

In Table 4, as shown in Fig. 5 above, the classification model was able to perform with a high level of accuracy in classification based on its ability to perform with an accuracy rate of 0.9789, a precision rate of 0.9753, a recall rate of 0.9793, an F1 score of 0.9773, and an ROC-AUC score of 0.9799 using five different metrics to measure its classification accuracy. Therefore, given the model's high classification accuracy, it can be used as a reliable component in the proposed adaptive security system. In addition, based on the precision and recall rates, it is possible to confirm that the model performs with a low false-positive rate while identifying most attacks as threats.

Table 4: Classification performance metrics of the proposed model.

Metric	Value
Accuracy	0.9789
Precision	0.9753
Recall	0.9793
F1-score	0.9773
ROC-AUC	0.9799

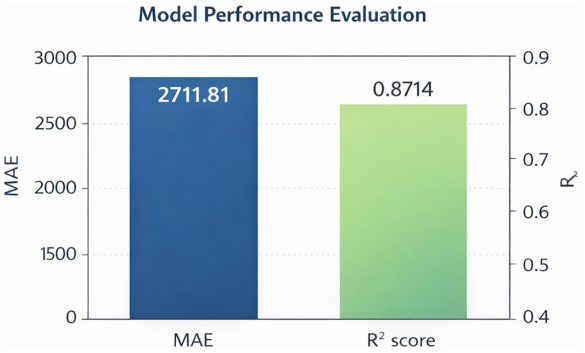

Figure 5: Model performance evaluation (classification metrics visualization).

4.2 Adaptive Security Policy Decision

Based on the computed risk score and the specified decision threshold, the adaptive security policy engine determined that the operational mode should be set to LOW security. This decision suggests that the classification output and the analysis of anomaly points point to a relatively low-risk system. Therefore, it does not require stringent control to be contained. Nevertheless, it should be noted that the LOW security policy does not imply that the system is insecure. Rather, it permits the framework to run in a usability-preserving mode while continuing to run the monitoring and logging processes. This helps to support the objective of preventing over-securitization while maintaining security awareness. The decision-making process of the adaptive security policy is depicted by Algorithm 1. The policy-aware incident response behavior in the HIGH and LOW security modes is visually represented in Fig. 6.

Algorithm 1: Adaptive security policy selection

Input: MeanAttackProbability p, AnomalyRate a, RiskThreshold τ
Output: SecurityPolicy
1: Compute RiskScore = 0.7 × p + 0.3 × a
2: if RiskScore ≥ τ then
3: SecurityPolicy←"HIGH."
4: else
5: SecurityPolicy←"LOW."
6: end if
7: return SecurityPolicy

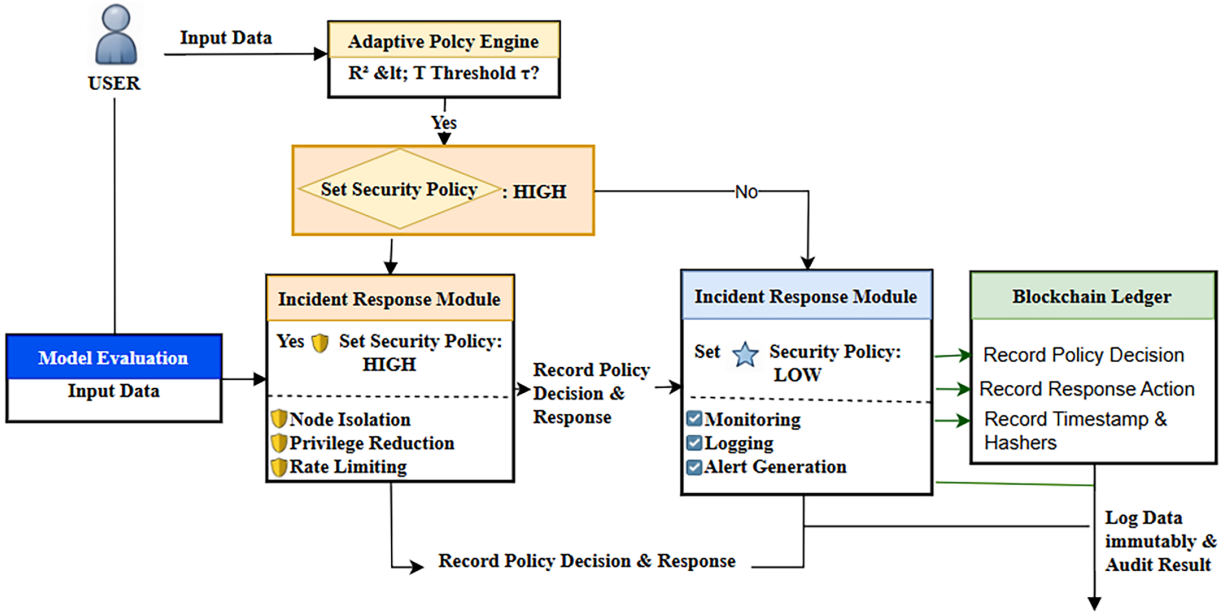

Figure 6: Policy-aware incident response under HIGH and LOW security modes.

Fig. 6 presents the risk-driven process of the adaptive security framework's incident response in HIGH and LOW security modes. The adaptive security framework's risk-driven process begins with classification and anomaly analysis. Here, the adaptive policy engine calculates a risk score based on the predicted probability of an attack and anomaly rate. This risk score is then compared with a threshold. If the threshold conditions are true, the system moves to HIGH security mode; otherwise, it stays in LOW security mode. Once in HIGH security mode, the incident response module initiates proactive response actions, such as node isolation, privilege reduction, and rate limiting, to address potential attacks. On the other hand, when in LOW security mode, the system allows passive response actions such as monitoring, logging, and alert generation. This ensures the system's usability. Both modes also ensure that the selected security policy and the corresponding response actions are recorded in a lightweight, tamper-evident, hash-linked ledger. This ensures traceability and accountability in recording security-related events and decisions.

4.3 Real-Time Anomaly Detection Results

The Isolation Forest-based anomaly detection module was used to detect anomalies in the test data. The results are binary: 1 indicates normal activity, and −1 indicates anomalous activity. The presence of both normal and anomalous data points in the detection results indicates that the anomaly detection system is actively monitoring system activities and is not in a dormant state even when LOW security conditions exist.

Fig. 7 shows the distributions of normal and anomalous events, derived from the anomaly detection results. From Fig. 7, it is evident that the proposed Isolation Forest-based module can identify both normal and anomalous traffic patterns during evaluation. This further demonstrates that the system's anomaly detection component has provided additional security monitoring functionality that was not present during the classification process. The anomaly detection results were also used in the proposed risk-aware policy mechanism.

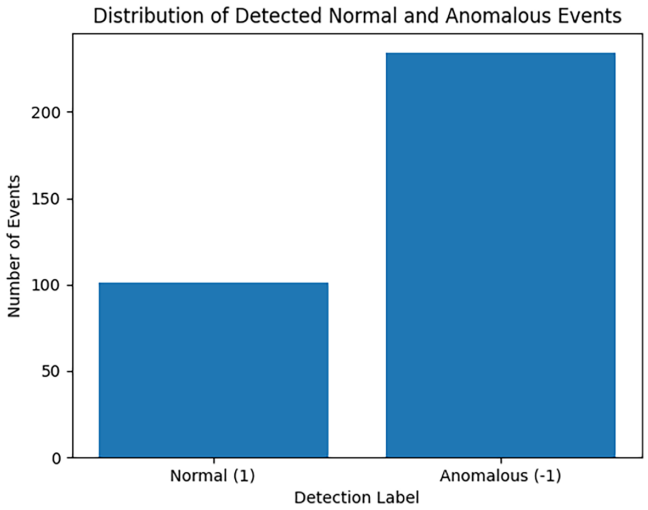

Figure 7: Distribution of detected normal and anomalous events.

4.4 Blockchain Ledger and Auditability Analysis

All critical system outputs, such as the dataset hash, model hash, classification performance metrics, risk scores, anomaly rate, chosen security policy, and response actions, are stored in a lightweight, tamper-evident, hash-linked ledger. The created ledger has a valid chain structure that starts with a genesis block and continues with subsequent blocks linked by cryptographic hash pointers. Any alteration in a pre-existing block would break the associated hash link, ensuring the integrity, traceability, and accountability of security-related decisions. This way, the proposed framework extends traditional security monitoring to an auditable and transparent security architecture.

Fig. 8 shows the design of the lightweight tamper-evident hash-linked ledger used in the proposed framework. The design starts with a Genesis Block, followed by subsequent blocks linked together by a cryptographic hash pointer. Each block contains the dataset hash, the model hash, the classification metrics, the risk score, the anomaly rate, the designated security policy, and the incident response. Chaining enables integrity, traceability, and tamper evidence, preserving security-related information throughout evaluation cycles.

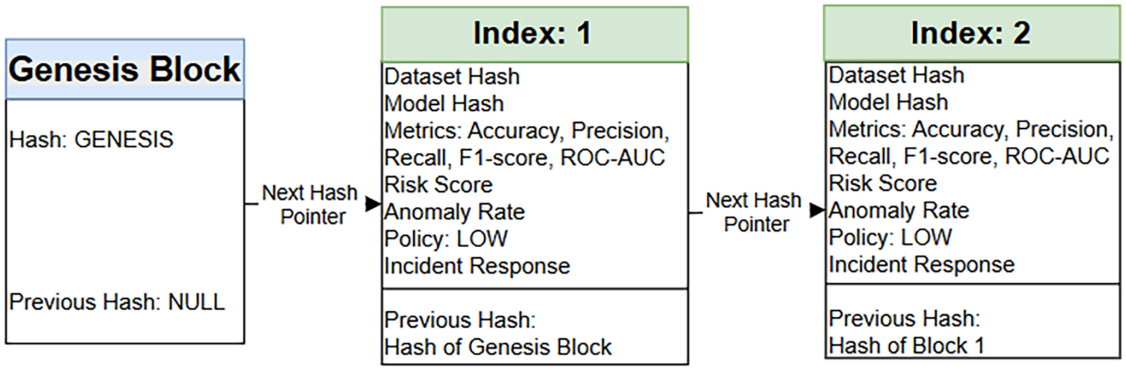

Figure 8: Lightweight tamper-evident hash-linked ledger structure for security decision records.

4.5 Usability Evaluation and Feedback Impact

Usability testing was conducted to assess system accessibility for non-expert users. The observed usability metrics are summarized as follows:

- **Total steps**: 4
- **Total errors**: 0
- **Setup time**: 4.63 s

The absence of user errors and the small setup time indicate that the system interface and interaction process are user-friendly. Based on the above parameters, the user satisfaction score falls in the average range. Therefore, the usability feedback module concluded that there was no need to relax or tighten the security policy, and the current LOW security policy was retained. This shows that the proposed framework achieves a balance between security and usability without undermining either.

The results obtained from the preliminary simulated usability evaluation are shown in Table 5. From the results, the usability indicators show that the framework needed only a few interaction steps, did not produce any user errors, and had a short setup time. These are desirable properties for a system that needs to be accessible to non-expert users. Based on usability feedback, the existing security policy remains unchanged at LOW.

Table 5: Usability metrics summary.

Metric	Value
Total Steps	4
Total Errors	0
Setup Time	4.63 s
Usability Feedback Level	Average
Final Security Policy	LOW

4.6 Integrated System Behavior Discussion

Experimental results demonstrate that the proposed framework can function as a closed-loop adaptive security system. This is because the classification component provides reliable discrimination between benign and malicious traffic, the anomaly detection component enables continuous monitoring of suspicious activity, and the adaptive policy component enables security responses to be generated based on risk-aware

evidence. At the same time, the lightweight tamper-evident hash-linked ledger ensures that security decisions and response actions are traceable. In this regard, unlike traditional static security systems that cannot adapt to threat conditions or usability-related interaction patterns, this proposed approach can adapt to both. This is particularly important in environments with non-expert users, as security restrictions may negatively impact usability and acceptance. Therefore, as demonstrated by the results obtained with this proposed approach, it is practical because it enables the integration of attack classification, anomaly-aware monitoring, auditable logging, and usability-aware adaptation within a unified adaptive security framework.

5 Conclusion

In this paper, an adaptive security framework based on blockchain technology was proposed, comprising machine-learning-based attack classification, anomaly detection, policy-based incident response systems, tamper-evident hash-linked logging systems, and usability feedback in a closed loop. In contrast to traditional security frameworks that rely on static security policies and auditing systems, the proposed security framework dynamically enforces security policies based on risk evidence and user interactions. An experimental evaluation using the CICIDS2017 benchmark intrusion detection system dataset was presented to demonstrate the reliability of the supervised learning model for binary classification of benign and malicious traffic flows. Furthermore, the anomaly detection module monitored flow-level behavior and detected suspicious patterns during evaluation. All critical decisions made during the experiment, including model performance, anomaly detection results, security policy, and incident response, were recorded using a lightweight, tamper-evident hash-linked ledger. In addition to the technical evaluation, the framework also considered its usability for non-experts, which is sometimes overlooked in security systems. The framework's usability was also considered, which is sometimes overlooked in security systems. The usability assessment of the framework revealed low interaction complexity, no user error, and a short setup time during evaluation. The framework thus illustrates its potential to strike a balance between security strength and user accessibility through security decision-making informed by usability. Overall, the proposed framework demonstrates its potential to combine adaptive intelligence, flow-level anomaly monitoring, tamper-evident audit logging, and usability-awareness into a unified, trustworthy security solution. Moreover, the results demonstrate the potential of the proposed framework in cybersecurity-related settings where security strength, reliability, traceability, and user accessibility are important factors.

6 Future Work

Although this proposed framework has demonstrated promising results, several avenues remain for further enhancing this work. First, future work could incorporate additional supervised learning models to further enhance the classification's robustness. Second, although this study has demonstrated effective anomaly detection using a single Isolation Forest algorithm, future work could incorporate hybrid models to better discriminate anomalies and reduce false alarms. Third, while this study relies on a lightweight tamper-evident hash-linked ledger, future studies may extend this framework to permissioned blockchain platforms to facilitate multi-node validation, trust management, and security coordination. Fourth, while this study relies on preliminary interaction-based indicators to assess usability, future studies may use formal human-subjects studies to assess the usability of such systems, employing frameworks such as SUS and NASA-TLX, as well as long-term studies and behavioral modeling to understand how non-expert users interact with adaptive security systems. Lastly, the proposed framework may be extended to various domains to assess its validity.

Acknowledgement: Not applicable.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: The authors confirm their contributions to the paper as follows: Mosammat Jannatul Kobra and Muhammad Rashid Majeed contributed to the conceptualization of the research. Mosammat Jannatul Kobra worked on the methodology, software development, formal analysis, investigation, and data curation. She also prepared the original draft of the manuscript and was involved in writing and editing the review. Muhammad Rashid Majeed supervised the project and contributed to the validation, review, and editing of the manuscript, as well as project administration. Md Owahedur Rahman contributed to the validation and review of the manuscript. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The processed data and models used in this research are available from the corresponding author upon reasonable request.

Ethic Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Chen H, Ali Babar M. Security for machine learning-based software systems: a survey of threats, practices, and challenges. ACM Comput Surv. 2024;56(6):1–38. doi:10.1145/3638531.
2. Conti M, Dehghantanha A, Franke K, Watson S. Internet of Things security and forensics: challenges and opportunities. Future Gener Comput Syst. 2018;78(6):544–6. doi:10.1016/j.future.2017.07.060.
3. Sommer R, Paxson V. Outside the closed world: on using machine learning for network intrusion detection. In: Proceedings of the 2010 IEEE Symposium on Security and Privacy; 2010 May 16–19; Oakland, CA, USA. p. 305–16. doi:10.1109/sp.2010.25.
4. Buczak AL, Guven E. A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Commun Surv Tutor. 2016;18(2):1153–76. doi:10.1109/comst.2015.2494502.
5. Squarepants S. Bitcoin: a peer-to-peer electronic cash system. [cited 2026 Jan 1]. Available from: https://ssrn.com/abstract=3440802.
6. Crosby M, Pattanayak P, Verma S, Kalyanaraman V. Blockchain technology: beyond bitcoin. Appl Innov. 2016;2(6–10):71.
7. Liu FT, Ting KM, Zhou ZH. Isolation forest. In: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining; 2008 Dec 15–19; Pisa, Italy. p. 413–22. doi:10.1109/icdm.2008.17.
8. Sasse MA, Brostoff S, Weirich D. Transforming the 'weakest link'—a human/computer interaction approach to usable and effective security. BT Technol J. 2001;19(3):122–31. doi:10.1023/A:1011902718709.
9. Furnell S, Clarke N. Power to the people? The evolving recognition of human aspects of security. Comput Secur. 2012;31(8):983–8. doi:10.1016/j.cose.2012.08.004.
10. Beautement A, Sasse MA, Wonham M. The compliance budget: managing security behaviour in organisations. In: Proceedings of the 2008 New Security Paradigms Workshop; 2008 Sep 22–25; Olympic Valley, CA, USA. p. 47–58. doi:10.1145/1595676.1595684.
11. Zhang R, Xue R, Liu L. Security and privacy on blockchain. ACM Comput Surv. 2020;52(3):1–34. doi:10.1145/3316481.
12. Mustafa R, Han J, Sarkar NI, Petrova K. Secure cross-layer mobile sensing framework for real-time disaster reporting and visualisation using a mobile application. Sensors. 2025;25(21):6766. doi:10.3390/s25216766.
13. Sharafaldin I, Habibi Lashkari A, Ghorbani AA. Toward generating a new intrusion detection dataset and intrusion traffic characterization. In: Proceedings of the 4th International Conference on Information Systems Security and Privacy; 2018 Jan 22–24; Funchal, Portugal. p. 108–16. doi:10.5220/0006639801080116.
14. Xu X, Weber I, Staples M. Architecture for blockchain applications. Cham, Switzerland: Springer International Publishing; 2019. doi:10.1007/978-3-030-03035-3.
15. Kshetri N. Can blockchain strengthen the Internet of Things? IT Prof. 2017;19(4):68–72. doi:10.1109/mitp.2017.3051335.

16. Zaghloul E, Li T, Mutka MW, Ren J. Bitcoin and blockchain: security and privacy. IEEE Internet Things J. 2020;7(10):10288–313. doi:10.1109/jiot.2020.3004273.
17. Androulaki E, Barger A, Bortnikov V, Cachin C, Christidis K, De Caro A, et al. Hyperledger fabric: a distributed operating system for permissioned blockchains. In: Proceedings of the 13th EuroSys Conference; 2018 Apr 23–26; Porto, Portugal. doi:10.1145/3190508.3190538.
18. Jain AK, Gupta N, Gupta BB. A survey on scalable consensus algorithms for blockchain technology. Cyber Secur Appl. 2025;3(4):100065. doi:10.1016/j.csa.2024.100065.
19. Chandola V, Banerjee A, Kumar V. Anomaly detection: a survey. ACM Comput Surv. 2009;41(3):1–58. doi:10.1145/1541880.1541882.
20. Li Z, Zhu Y, Van Leeuwen M. A survey on explainable anomaly detection. ACM Trans Knowl Discov Data. 2023;18(1):1–54.
21. Xu H, Pang G, Wang Y, Wang Y. Deep isolation forest for anomaly detection. IEEE Trans Knowl Data Eng. 2023;35(12):12591–604. doi:10.1109/tkde.2023.3270293.
22. Maseer ZK, Yusof R, Bahaman N, Mostafa SA, Foozy CFM. Benchmarking of machine learning for anomaly based intrusion detection systems in the CICIDS2017 dataset. IEEE Access. 2021;9:22351–70. doi:10.1109/access.2021.3056614.
23. Iscenko Z, Pickard C, Smart L, Vasas Z. Behaviour and compliance in organisations. FCA occasional paper. 2016 [cited 2026 Jan 1]. Available from: https://ssrn.com/abstract=2939687.
24. Weir C, Dyson A, Prince D. Do you speak cyber? Talking security with developers of health systems and devices. IEEE Secur Privacy. 2023;21(1):27–36. doi:10.1109/msec.2022.3221616.
25. Tahaei M, Vaniea K, Rashid A. Embedding privacy into design through software developers: challenges and solutions. IEEE Secur Priv. 2023;21(1):49–57. doi:10.1109/msec.2022.3204364.
26. Xin Y, Kong L, Liu Z, Chen Y, Li Y, Zhu H, et al. Machine learning and deep learning methods for cybersecurity. IEEE Access. 2018;6:35365–81. doi:10.1109/access.2018.2836950.
27. Wang YC, Houng YC, Chen HX, Tseng SM. Network anomaly intrusion detection based on deep learning approach. Sensors. 2023;23(4):2171. doi:10.3390/s23042171.
28. Si-Ahmed A, Ali Al-Garadi M, Boustia N. Survey of machine learning based intrusion detection methods for internet of medical things. Appl Soft Comput. 2023;140(3):110227. doi:10.1016/j.asoc.2023.110227.
29. Memon M, Hussain SS, Bajwa UA, Ikhlas A. Blockchain beyond Bitcoin: blockchain technology challenges and real-world applications. In: Proceedings of the 2018 International Conference on Computing, Electronics & Communications Engineering (iCCECE); 2018 Aug 16–17; Southend, UK. p. 29–34. doi:10.1109/iccecome.2018.8658518.
30. Shrimali B, Patel HB. Blockchain state-of-the-art: architecture, use cases, consensus, challenges and opportunities. J King Saud Univ Comput Inf Sci. 2022;34(9):6793–807. doi:10.1016/j.jksuci.2021.08.005.
31. Dorri A, Kanhere SS, Jurdak R. Blockchain in internet of things: challenges and solutions. arXiv:1608.05187. 2016.
32. Brooke J. SUS: a “quick and dirty” usability scale. In: Jordan PW, Thomas B, Weerdmeester BA, McClelland IL, editors. Usability evaluation in industry. Abingdon, UK: Talylor Francis; 1996. p. 189–94.
33. Fazelnia M, Okutan A, Mirakhorli M. Supporting artificial intelligence/machine learning security workers through an adversarial techniques, tools, and common knowledge framework. IEEE Secur Priv. 2023;21(1):37–48. doi:10.1109/msec.2022.3221058.
34. Pattnaik N, Li S, Nurse JRC. Perspectives of non-expert users on cyber security and privacy: an analysis of online discussions on twitter. Comput Secur. 2023;125:103008. doi:10.1016/j.cose.2022.103008.
35. Di Nocera F, Tempestini G, Orsini M. Usable security: a systematic literature review. Information. 2023;14(12):641. doi:10.3390/info14120641.
36. Morice D. Trust framework on exploitation of humans as the weakest link in cybersecurity. Appl Cybersecur Internet Gov. 2023;2(1):184310. doi:10.60097/acig/162867.
37. Gorski PL, Iacono LL, Smith M. Eight lightweight usable security principles for developers. IEEE Secur Priv. 2023;21(1):20–6. doi:10.1109/msec.2022.3205484.