Annals of Computer Science and Intelligence Systems
https://doi.org/10.5281/zenodo.18481200

Tamper-Resistant Zero Trust Access Management with Intelligent Intrusion Detection

Sultan Algarni
Department of Information Systems, Faculty of Computing and Information Technology
King Abdulaziz University, Jeddah 21589, Saudi Arabia
saalgarni@kau.edu.sa

Received: Dec 29, 2025 **Accepted:** Jan 29, 2026 **Published:** Feb 04, 2026

Abstract

Modern enterprises increasingly rely on cloud services, mobile devices, and Internet-of-Things (IoT) assets, resulting in highly distributed and dynamic attack surfaces. Traditional perimeter-based security models and static access control schemes are no longer sufficient to protect critical resources against sophisticated adversaries. Zero Trust Architecture (ZTA) has emerged as a leading paradigm that enforces the principle of "never trust, always verify" by continuously evaluating user identity, device posture, and contextual risk [1, 2]. However, current ZTA deployments still rely on logically centralized policy decision points and conventional logging infrastructures, which raise concerns regarding trust, integrity, and resilience. In parallel, intrusion detection systems (IDS) struggle to keep pace with evolving attack techniques when they depend solely on signature-based or manually tuned rules [14, 15].

This paper proposes a blockchain-enabled zero trust access control framework augmented with intelligent, AI-driven intrusion detection for modern cybersecurity systems. The framework leverages a permissioned blockchain to record identities, access policies, and audit logs as tamper-evident, verifiable records, thereby decentralizing trust and strengthening accountability [7, 8]. Access control decisions are expressed and enforced through smart contracts, while a machine learning–based IDS continuously analyzes network and host telemetry to detect anomalies and high-risk behaviors [17, 18]. Detected threats dynamically influence access decisions via a risk-adaptive feedback loop. We empirically evaluate the intelligent IDS component using the UNSW-NB15 dataset, a modern benchmark for network intrusion detection research [19].

We describe the system and threat models, present the overall architecture, detail the machine learning methodology, and report experimental results on detection performance. A security and architectural discussion highlights how the proposed approach mitigates key attack vectors such as policy manipulation, log tampering, and stealthy lateral movement, illustrating the potential of combining blockchain, ZTA, and AI-driven IDS in next-generation cybersecurity systems.

Keywords—Blockchain security, Zero Trust Architecture (ZTA), access control models, intelligent intrusion detection systems (IDS), AI-driven intrusion detection, UNSW-NB15.

1 Introduction

The rapid adoption of cloud computing, mobile devices, and Internet-of-Things (IoT) technologies has transformed the way organizations design and operate their information systems. Business-critical workflows increasingly span multiple administrative domains and networks, including public clouds, on-premises data centers, software-as-a-service (SaaS) platforms, and remote endpoints [3, 4].

Adversaries routinely bypass perimeter defenses by exploiting software vulnerabilities, phishing credentials, abusing misconfigurations, or compromising third-party providers. Once inside,

they can perform stealthy lateral movement, privilege escalation, and data exfiltration, often remaining undetected for extended periods [1, 2, 18]. Access to resources is granted only after continuous verification of user identity, device posture, and contextual attributes such as geolocation, time of access, and behavioral signals.

While ZTA significantly improves security compared to legacy models, many current imple- mentations rely on logically centralized policy decision points (PDPs) and conventional logging infrastructures [4, 5]. This centralization introduces single points of failure and raises concerns about the integrity, availability, and auditability of access control decisions and security log [24]. An attacker who manages to tamper with policy stores or logging systems may be able to mis- represent access decisions, erase traces of malicious activity, or subvert compliance reporting. By anchoring identities, policies, and audit logs on a permissioned blockchain, our framework provides a tamper-resistant control plane in which unauthorized modifications to critical secu- rity data are significantly harder to perform and easier to detect [14, 15].

Although effective against previously observed threats, such systems often fail to detect zero-day exploits, polymorphic malware, and subtle lateral movement. The increasing volume, velocity, and variety of network and host telemetry further challenge traditional IDS designs, necessitating more scalable and intelligent approaches [17].

Recent advances in machine learning (ML) and artificial intelligence (AI) have enabled more capable IDS solutions that can model normal behavior, identify anomalies, and adapt to evolving attack techniques [20–22]. At the same time, blockchain technology has demonstrated strong guarantees of integrity, tamper-evidence, and decentralization, and has been explored for secure logging, identity management, and access control in distributed environments [7, 9–11]. Integrating these technologies within a coherent Zero Trust framework offers an opportunity to strengthen both the trust layer and the detection layer of modern cybersecurity systems.

This paper argues that combining blockchain technology, Zero Trust access control, and AI-driven intrusion detection can address several shortcomings of existing approaches. Blockchain's decentralization and immutability can enhance the trustworthiness and resilience of identity management, policy enforcement, and audit logging [8, 13]. Meanwhile, intelligent IDS can provide continuous, data-driven risk assessment that informs and adapts access control decisions. To assess the feasibility of this vision, we design and evaluate an ML-based IDS component using the UNSW-NB15 dataset, a comprehensive modern network intrusion detection benchmark [19].

The main contributions of this work are:

- We propose a blockchain-enabled Zero Trust access control framework in which identities, access policies, and audit logs are anchored on a permissioned blockchain, enabling tamper-evident and verifiable security controls.
- We integrate an AI-driven intrusion detection component that analyzes network flow fea- tures and contextual signals to detect malicious behavior and dynamically influence risk- adaptive access decisions.
- We present a machine learning methodology and experimental evaluation of the IDS component using the UNSW-NB15 dataset, and we discuss architectural trade-offs, limita- tions, and future research directions for end-to-end deployment.

The rest of this paper is structured as follows. Section 2 surveys related work on Zero Trust Architecture (ZTA), blockchain-based access control, and intelligent intrusion detection systems. Section 3 details the proposed architecture and methodology. Section 4 describes the experimental setup and reports the results obtained using the UNSW-NB15 dataset. Section 5 discusses the implications, advantages, and limitations of the proposed approach. Finally, Section 6 concludes the paper.

2 State of the Art

This section reviews related work in four main areas: Zero Trust Architectures, blockchain-based access control and logging, intelligent intrusion detection systems, and datasets for evaluating IDSs. We highlight key contributions and limitations, and identify research gaps that motivate our proposed framework.

2.1 Zero Trust Architecture

The Zero Trust security model was initially articulated by Kindervag in a foundational industry report that challenged the reliance on implicit trust derived from network perimeters [2]. Since then, the concept has been extensively expanded by both academia and industry into a comprehensive architectural paradigm. The U.S. National Institute of Standards and Technology (NIST) subsequently codified Zero Trust principles and reference architectures in Special Publication 800-207, highlighting continuous verification, least-privilege access, and policy-driven decision-making as core elements of the approach [1].

Recent surveys have analyzed ZTA concepts, deployment patterns, and challenges in large-scale environments [3,4,6]. These works highlight that, in practice, ZTA deployments often rely on centralized identity providers, policy engines, and data repositories, which may become attractive targets and single points of failure. Some authors have proposed federated or distributed approaches to mitigate these issues, but relatively few have explored the use of blockchains to provide decentralized and tamper-evident policy and logging layers [5].

Moreover, the integration of advanced analytics and machine learning into ZTA decision-making remains a developing area. While many commercial ZTA solutions incorporate risk scores or behavioral analysis in access decisions, the underlying models and their systemic integration are often proprietary and not rigorously evaluated in the literature [6].

2.2 Blockchain for Access Control and Logging

Blockchain technology, introduced with Bitcoin as a decentralized ledger for digital currency [7], has been widely studied for its integrity, transparency, and fault tolerance. Surveys on blockchain security and applications underline its potential for secure logging, audit trails, and distributed access control in multi-stakeholder environments [9, 13, 25].

Permissioned blockchain platforms such as Hyperledger Fabric support configurable membership, pluggable consensus, and smart contracts, enabling enterprise-oriented access control solutions [8]. Several works have proposed blockchain-based access control mechanisms for IoT, cloud, and data-sharing scenarios, where policies and capabilities are encoded as transactions or smart contracts [10–12]. These schemes aim to eliminate centralized policy stores and provide tamper-evident records of authorization decisions and resource usage.

However, many existing blockchain-based access control proposals focus on static or coarse-grained policies, and often do not integrate real-time risk assessment or intrusion detection signals into the policy evaluation process [10, 12, 25]. Additionally, scalability, latency, and pri- vacy concerns limit the direct use of blockchains for high-volume, low-latency security telemetry streams.

2.3 Intrusion Detection Systems and Machine Learning

Denning's pioneering work defined an intrusion-detection model based on anomaly detection and audit data analysis [14]. Subsequent efforts classified IDSs into signature-based (misuse detection) and anomaly-based systems, as well as into host-based and network-based deployments [15, 16].

With the increasing complexity of network environments and attacks, machine learning has become a central tool in IDS research. Surveys and empirical studies have compared classical

ML techniques and deep learning approaches (e.g., autoencoders, convolutional and recurrent neural networks) for intrusion detection tasks [17,20–22]. These works generally report improved detection rates and generalization compared to purely signature-based systems, especially for previously unseen attacks.

Nevertheless, several challenges remain. Anomaly-based methods can suffer from high false-positive rates, concept drift in dynamic environments, and difficulties in explaining model decisions to human analysts [18]. Moreover, many ML-based IDS proposals are evaluated on dated or unrealistic datasets, limiting their relevance to contemporary threats and network conditions [17].

2.4 Intrusion Detection Datasets

The availability of representative datasets is crucial for designing, training, and evaluating IDSs. Early benchmark datasets such as KDD'99 and its variants have been widely used but criticized for multiple issues, including redundancy, unrealistic traffic patterns, and outdated attack scenarios [17].

UNSW-NB15 is a comprehensive dataset generated in a controlled testbed, combining realistic benign traffic with modern synthetic attack scenarios [19]. It includes packet captures and derived flow features, along with binary labels (normal vs. attack) and multi-class attack categories. Subsequent studies have used UNSW-NB15 to evaluate and compare various ML-based IDSs, including deep learning models [20, 21, 23].

Despite these advances, there is still a need to better integrate dataset-driven IDS evaluation with broader architectural considerations such as Zero Trust and blockchain-based control planes. Most prior works either focus on IDS accuracy in isolation or treat blockchain/ZTA components only conceptually without a concrete data-driven evaluation of detection performance.

In summary, the literature shows active research on Zero Trust architectures [1,3,4], blockchain- based access control and logging [8–12], and ML models [17, 19–22]. However, there is compar- atively little work on tightly integrating these components into a coherent, risk-adaptive Zero Trust framework that leverages blockchain for trustworthy policy and logging, and AI-based IDS for continuous risk assessment [5, 6, 23].

Our work addresses this gap by (i) proposing an end-to-end architecture that combines a permissioned blockchain with Zero Trust access control, and (ii) implementing and empirically evaluating an intelligent IDS component using the UNSW-NB15 dataset, explicitly linking detection outcomes to dynamic access control decisions.

3 Proposed Methodology

This section presents the overall methodology, including the system model, the blockchain-enabled Zero Trust access control design, and the intelligent intrusion detection component. We focus on the logical architecture and the interactions among components rather than on low-level implementation details.

3.1 System Model

We consider a modern enterprise environment in which users, devices, and services are geographically distributed and connected through heterogeneous networks (corporate LANs, Wi-Fi, VPNs, public clouds, and partner networks). The organization operates according to Zero Trust principles: no implicit trust is granted based on network location, and every access to a protected resource must be explicitly authorized and continuously re-evaluated.
The system comprises the following main entities (also illustrated in Fig. 1):

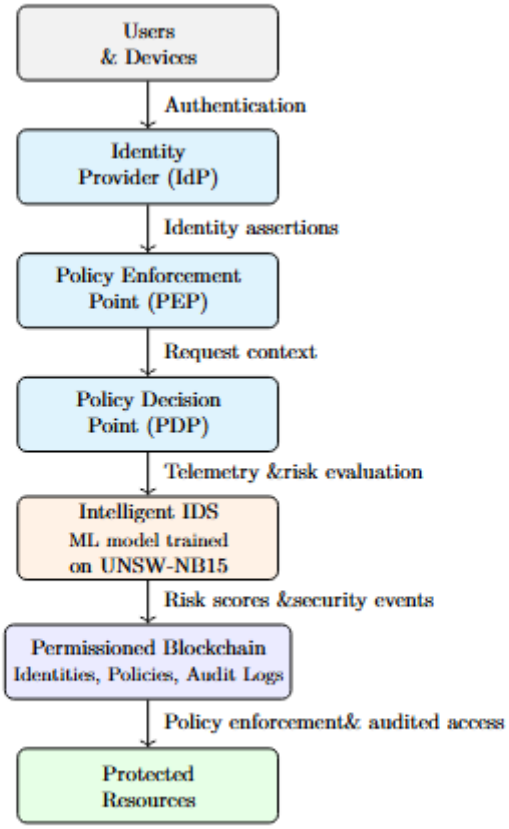

Figure 1: Proposed blockchain-enabled Zero Trust methodology

- **Users and devices** (subjects): human users (employees, contractors, partners) and their associated endpoints (laptops, mobile devices, virtual machines, IoT devices). Each sub- ject is represented by a digital identity and a set of attributes (e.g., role, department, device posture). Devices may be corporate-managed or bring-your-own-device (BYOD), but access requirements are typically stricter for unmanaged endpoints.
- **Protected resources** (objects): applications, microservices, APIs, databases, file shares, and other assets that expose interfaces over the network. Resources can be hosted in on-premises data centers, private clouds, or public cloud platforms. Each resource is associated with access control policies and sensitivity levels.
- **Identity Provider (IdP)**: an authentication and identity management service (e.g., based on SAML, OpenID Connect, or LDAP) that verifies user credentials, manages device registrations, and issues signed identity assertions containing subject identifiers and attributes. The IdP also participates in lifecycle management operations such as onboarding, offboarding, and credential revocation.
- **Policy Enforcement Point (PEP)**: a logical component deployed at the boundary of each protected resource (e.g., API gateway, reverse proxy, sidecar, host agent). The PEP intercepts incoming requests, extracts the relevant context (identity assertions, de- vice posture, resource identifier, action, environmental attributes), and consults the Policy Decision Point to obtain an authorization decision (e.g., allow, deny, step-up authentica- tion, limited access).
- **Policy Decision Point (PDP) / Policy Engine**: the central decision-making compo- nent in the Zero Trust control plane. The PDP receives normalized request contexts from PEPs, retrieves the applicable policies and subject attributes, incorporates the latest risk information from the IDS, and computes the final decision according to the organization's Zero Trust policies. The

PDP interacts with the blockchain to obtain tamper-evident policy and identity records and to record audit-relevant events.

- **Permissioned blockchain network**: a consortium-style distributed ledger operated by the organization and, optionally, selected trusted partners. It runs a permissioned blockchain platform (e.g., Hyperledger Fabric) in which validator nodes are authenticated and authorized. The blockchain stores:
  - identity anchors and selected attributes (or cryptographic commitments to off-chain identity records),
  - access control policies, represented in a structured format and possibly enforced by smart contracts,
  - audit logs of significant security events (e.g., access decisions, policy updates, high- severity alerts).

  The blockchain provides integrity, tamper-evidence, and non-repudiation guarantees for these records.
- **Intelligent Intrusion Detection System (IDS)**: an AI-driven analysis component that processes security telemetry (primarily network flow features in this work) and outputs both binary intrusion predictions and continuous risk scores. The IDS is trained offline on UNSW-NB15. At runtime, it consumes flow/context information from PEPs and re- sources and periodically updates risk assessments associated with subjects (users/devices), sessions, or network segments.

The logical architecture separates the data plane (actual traffic between users/devices and resources) from the control plane (identity, policy, risk, and logging decisions):

- In the **data plane**, users and devices initiate connections to protected resources through PEPs. The PEPs enforce access decisions returned by the PDP and may also perform local rate limiting, protocol normalization, or basic input validation.
- In the **control plane**, the IdP authenticates subjects and issues signed tokens; the PDP evaluates Zero Trust policies using identity and policy information anchored on the blockchain and risk scores supplied by the IDS; the blockchain provides a shared, tamper- evident store for critical security metadata and audit trails; and the IDS continuously refines its internal models and risk estimates based on observed telemetry.

We assume that communications between these components are secured using mutually authenticated and encrypted channels (e.g., TLS with client certificates or modern service mesh solutions), and that each component has access to a hardware or software-based root of trust for key management. The blockchain nodes are distributed across independent administrative domains within the organization (e.g., different business units or data centers), so that no single compromised node can unilaterally rewrite history or suppress security events without detection.
From a temporal perspective, the system model distinguishes between two phases:

1. **Design and provisioning phase**:
   - Identities are created, and attribute schemas are defined.
   - Access control policies are authored by security administrators and deployed as on- chain policy records or smart contracts.
   - Blockchain nodes are provisioned, and consensus parameters are configured.
2. **Operational phase**:
   - For each access attempt, the PEP extracts the context, forwards it to the PDP, and enforces the returned decision.
   - The PDP queries on-chain identity and policy records, retrieves the current risk score for the requesting subject from the IDS, and combines these inputs to produce a Zero Trust decision.
   - The resulting decision and relevant metadata (e.g., subject ID, resource ID, time, risk score, decision outcome) are written to the blockchain as an audit event.

- The IDS continuously ingests flow and log data exported by PEPs and resources, updates its predictions and risk scores, and optionally emits alerts; selected high- level alerts and aggregated risk indicators are also anchored on the blockchain.

Within this model, trust in access control and logging does not depend on any single central database or log server. Instead, the combination of a permissioned blockchain and an intelligent IDS provides (i) robust integrity and accountability for identity, policy, and audit data, and (ii) dynamic, behavior-based risk assessment that can be fed back into Zero Trust policies. The subsequent subsections detail how access control is implemented on top of the blockchain and how the IDS is designed, trained, and integrated into the control plane.

3.2 Blockchain-Enabled Zero Trust Access Control

In our design, the blockchain serves three main purposes: (i) as a secure registry of identities and attributes, (ii) as a policy store and enforcement substrate via smart contracts, and (iii) as an immutable audit log of access decisions and security-relevant events.

Identity and Attribute Management: User and device identities, along with selected attributes (e.g., roles, departments, device compliance status), are anchored on the blockchain as signed records. Updates to identities (e.g., onboarding, revocation, attribute changes) are recorded as transactions, ensuring a verifiable history of identity lifecycle events. The blockchain does not necessarily store sensitive personally identifiable information (PII) directly; instead, it can store pseudonymous identifiers and hashed references to off-chain identity data, preserving privacy while ensuring integrity.

Policy Representation and Smart Contracts Access control policies are expressed as conditions over identities, attributes, resource types, and contextual signals (e.g., time, location, risk level). The smart contracts reference on-chain identity records and policy rules, as well as a risk score maintained by the IDS component. Policy evaluation results (e.g., allow, deny, require step-up authentication) are returned to the PDP and recorded as audit events on the blockchain.

Audit Logging: All significant security events, including successful and failed access attempts, policy changes, and security alerts, are recorded on the blockchain as append-only logs. This provides a tamper-evident, time-stamped trail that can be used for forensic analysis, compliance audits, and anomaly detection. To address scalability and privacy, high-volume raw logs can be stored off-chain, with cryptographic hashes anchored on-chain to ensure integrity and non- repudiation.

3.3 Intelligent Intrusion Detection

We focus on a network-based IDS (NIDS) trained on the UNSW-NB15:

Data Sources In a real deployment, the IDS would ingest network flow records, packet metadata, endpoint telemetry, and possibly application logs. For this study, we concentrate on flow-based features analogous to those provided in UNSW-NB15 [19]. These features capture statistics such as connection duration, bytes sent/received, packet counts, and selected protocol flags.

Machine Learning Model We frame intrusion detection as a supervised binary classification problem (normal vs. malicious), with the option to extend to multi-class attack categorization. Let $x \in \mathbb{R}^d$ denote a feature vector derived from a network flow, and $y \in \{0, 1\}$ the corresponding label (0 = normal, 1 = attack). Our goal is to learn a function $f_\theta : \mathbb{R}^d \to [0, 1]$ parameterized by θ, that estimates the probability that a given flow is malicious:

$$\hat{p}(y = 1 \mid x) = f_\theta(x).$$

In our experiments, we consider tree-based ensemble models such as Random Forests and Gradient Boosted Trees, which have shown strong performance on tabular intrusion detection

data [17, 23]. These models can handle heterogeneous feature types and non-linear interactions, and provide reasonable interpretability.

Risk Scoring and Integration with ZTA The IDS outputs both a binary prediction and a continuous anomaly or risk score r ∉0, 1]. We use the risk score as an input attribute in the ZTA policy evaluation. For example, policies may specify that:

- For low-risk requests ($r < 0.3$), grant access if baseline ABAC conditions are satisfied.
- For medium-risk requests ($0.3 \leq r < 0.7$), require step-up authentication or limit access to non-sensitive resources.
- For high-risk requests ($r \geq 0.7$), deny access and trigger incident response workflows. The current risk score and selected IDS alerts for a given entity (e.g., user, device, or IP address) are periodically anchored on the blockchain, enabling policies to reference these attributes in a decentralized and verifiable manner. This creates a feedback loop in which detection outcomes directly influence access control decisions, while access logs provide additional context for detection.

4 Experimental Evaluation

This section evaluates the intelligent intrusion detection component of our framework on the UNSW-NB15 dataset. Our aims are to (i) assess how well a machine-learning-based IDS can distinguish malicious from benign flows in a realistic setting, and (ii) show that the model's continuous output scores are suitable for driving risk-adaptive Zero Trust policies.

4.1 Evaluation Objectives

We focus on the following evaluation objectives:

- **Detection effectiveness**: measure how accurately the model classifies flows as benign or malicious.
- **Error trade-offs**: analyze the balance between detection rate (recall) and false positives, as relevant for Zero Trust enforcement.
- **Risk scoring suitability**: validate that the model's probability outputs can be inter- preted as meaningful risk scores for Zero Trust policy decisions.

4.2 Dataset Overview

In this work, we rely on the official training and test splits delivered as CSV files within the Training and Testing package. Each record corresponds to a network flow, described by 49 features and two labels: a binary label (0 = normal, 1 = attack) and a categorical attack_cat specifying the attack type for malicious samples.

Table 1 summarizes the dataset splits used in our experiments. The exact counts are obtained programmatically and can be reproduced from the public CSV files.

Table 1: Overview of the UNSW-NB15 dataset splits used in our experiments.

Split	Total records	Benign	Malicious	Malicious (%)
Training set	175 341	56 000	119 341	68.06
Test set	82 332	37 000	45 332	55.06

The dataset's features can be roughly grouped into four categories, as shown in Table 2. This grouping follows prior work on UNSW-NB15 and similar flow-based intrusion detection datasets [17].

Table 2: Main feature categories in UNSW-NB15 (non-exhaustive examples).

Category	Example features
Basic connection	dur (duration), proto (protocol), service, state
Content / payload	sbytes, dbytes, spkts, dpkts, smean, dmean
Time-based stats	sttl, dttl, sloss, dloss, synack, ackdat
Connection behavior	ct_state_ttl, ct_srv_src,ct_srv_dst, ct_dst_src_ltr, s_ftp_login

4.3 IDS Pipeline

Figure 2 gives an overview of the IDS workflow applied to UNSW-NB15, from raw CSV files to risk scores integrated into the Zero Trust access control layer.

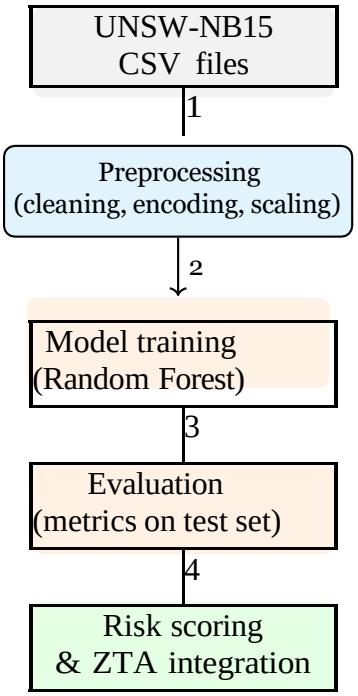

Figure 2: Workflow of the intelligent IDS component (vertical view). (1) UNSW-NB15 training and test CSV files are loaded; (2) features are preprocessed (cleaning, categorical encoding, normalization); (3) a Random Forest model is trained on the training split; (4) the model is evaluated on the test split and produces probability scores that are used as risk scores in the Zero Trust access control layer.

4.4 Preprocessing Pipeline

We apply a uniform preprocessing pipeline to both training and test sets to obtain suitable feature matrices for machine learning:

1. **Label selection**: we use the binary label column, mapping 0 to benign and 1 to mali- cious traffic.
2. **Feature selection**: we remove purely administrative or index-like fields (e.g., record identifiers) if present, and retain all informative numeric and categorical features, following guidelines from [17].
3. **Handling missing and invalid values**: any missing entries are imputed using appropri- ate strategies (median for numerical features, most frequent value for categorical features). Invalid or out-of-range values are either corrected or discarded, depending on frequency and impact.

4. **Normalization**: numerical features are standardized to zero mean and unit variance using statistics computed on the training set only. The same scaling parameters are then applied to the test set.

4.5 Model Configuration

We adopt a Random Forest classifier as a strong baseline for tabular intrusion detection data [17, 23].

Let $D_{train} = \{(x_i, y_i)\}_{i=1}^{N}$ denote the preprocessed training dataset, where $x_i \in R^d$ is a feature vector and $y_i \in \{0, 1\}$ the label. The Random Forest model consists of K decision trees
$\{T_k\}$, and the predicted probability that a flow is malicious is:

$$\hat{p}(y = 1 \mid x) = \frac{1}{K} \sum_{k=1}^{K} T_k(x).$$

We use this probability $\hat{p}(y = 1 \mid x)$ as the risk score $r \in [0, 1]$ in our Zero Trust architecture.

The main hyperparameters of our Random Forest configuration are summarized in Table 3. Hyperparameters are tuned empirically to achieve a good balance between detection performance and training time.

Table 3: Random Forest hyperparameters used in our experiments.

Hyperparameter	**Value**
Number of trees (K)	200
Maximum tree depth	20
Minimum samples per leaf	5
Criterion	Gini impurity
Max. Features per split	$\sqrt{\#\text{features}}$
Class weight	balanced (to handle class imbalance)
Random seed	42

Other models (e.g., Gradient Boosted Trees, XGBoost, LightGBM, or deep learning archi- tectures) could also be evaluated, but we focus on Random Forests for clarity and computational efficiency.

4.6 Results and Analysis

Table 4 reports the main evaluation metrics of the Random Forest classifier on the UNSW-NB15 test set for the binary intrusion detection task. The model achieves an accuracy of 0.9059, a precision of 0.8734, a recall of 0.9696, an F1-score of 0.9190, and a ROC-AUC of 0.9846.

Table 4: Classification performance of the Random Forest IDS on the UNSW-NB15 test set (binary classification).

Model	**Accuracy**	**Precision**	**Recall**	**F1-score**	**ROC-AUC**
Random Forest (RF)	0.9059	0.8734	0.9696	0.9190	0.9846

Table 5 summarizes the confusion matrix. The model correctly identifies the majority of both benign and malicious flows, but exhibits a non-negligible number of false positives and false negatives.

Figure 3 provides a graphical view of the same confusion matrix using a simple 2x2 grid.

Because our model outputs a continuous probability score $\hat{p}(y = 1 \mid x)$ for each flow, we can adjust the decision threshold $\tau \in [0, 1]$ for classifying a flow as malicious:

$$\text{predict malicious} \iff \hat{p}(y = 1 \mid x) \geq \tau.$$

Table 5: Confusion matrix

	Predicted class	
True class	**Benign**	**Malicious**
Benign	TN = 30628	FP = 6372
Malicious	FN = 1377	TP = 43955

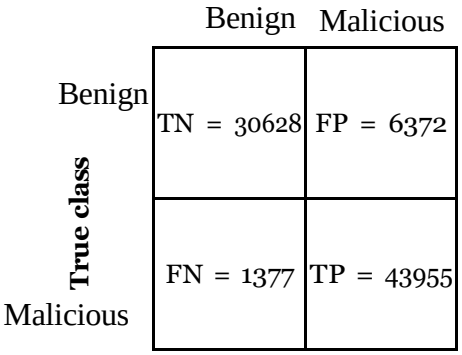

Figure 3: Confusion matrix of the Random Forest IDS on the UNSW-NB15 test set.

Table 6 illustrates how varying the threshold affects precision and recall. At a low threshold $\tau = 0.30$, the model achieves a very high recall of 0.9960 at the expense of precision (0.7800), which corresponds to a highly aggressive detection regime. At $\tau = 0.50$, precision and recall are more balanced (0.8734 and 0.9696), while at $\tau = 0.70$ the model reaches a precision of 0.9472 with a recall of 0.9171, corresponding to a stricter, low–false-positive regime.

Table 6: Operating points for different decision thresholds on the RF risk score.

Threshold τ	**Precision**	**Recall**	**Comment**
0.30	0.7800	0.9960	High recall, more false positives
0.50	0.8734	0.9696	Balanced operating point
0.70	0.9472	0.9171	Fewer false positives, lower recall

Figure 4 presents the ROC curve of the Random Forest model, generated from the same experiments. The area under the ROC curve (AUC) is 0.9846, which indicates excellent discrimination between benign and malicious flows across a wide range of thresholds.

Overall, the results demonstrate that the Random Forest model achieves high detection rates and excellent ROC-AUC on UNSW-NB15, in line with or exceeding prior studies on this dataset [17, 23]. The continuous probability outputs provide a natural and interpretable notion of risk that can be integrated into Zero Trust policies. Specifically, the PDP can use the model's risk score as a contextual attribute when evaluating access requests, tightening or relaxing access based on the current assessed threat level, while the blockchain layer records summarized detection outcomes and high-severity alerts as tamper-evident audit events.

5 Discussion

The experimental evaluation demonstrates that an ML-based IDS trained on UNSW-NB15 can achieve strong detection performance, supporting its use as a risk assessment engine within a Zero Trust framework. In this section, we discuss the broader implications, benefits, and limitations of integrating such an IDS with blockchain-enabled access control.

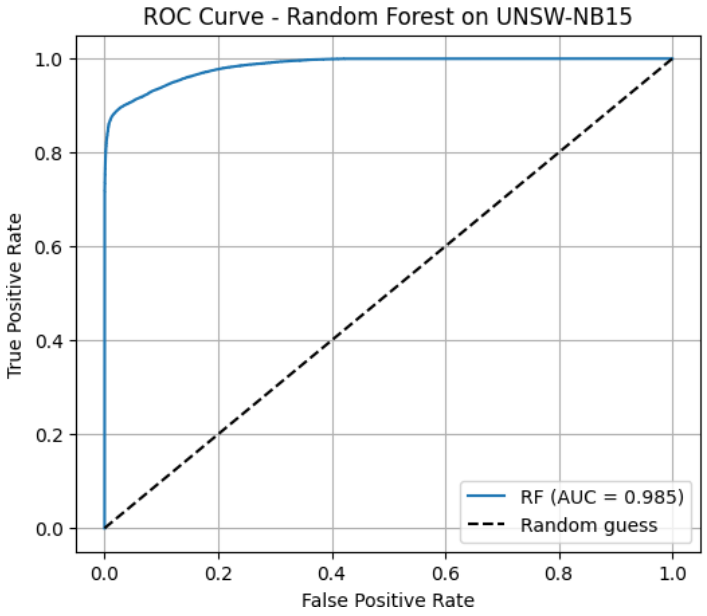

Figure 4: ROC curve of the Random Forest IDS on the UNSW-NB15 test set.

5.1 Security Benefits

By anchoring identities, policies, and audit logs on a

permissioned blockchain, our framework reduces the risk of undetected policy manipulation and log tampering. Even if an attacker compromises a single administrative domain or server, they cannot unilaterally alter historical records or covertly modify access policies without detection, as such changes require consensus among blockchain validators.

The integration of IDS risk scores into ZTA policies enables dynamic, context-aware access control. Rather than treating detection and authorization as separate silos, our approach uses IDS outputs as first-class inputs to policy evaluation. This can limit the blast radius of successful intrusions by restricting privileges for entities exhibiting suspicious behavior, and by forcing re- authentication or additional verification for medium-risk activities.

5.2 Performance and Scalability Considerations

Using a blockchain for security metadata introduces performance and scalability challenges. Directly recording every low-level event or IDS alert on-chain may lead to excessive transaction volumes and latency. To mitigate this, we advocate a layered approach in which:

- High-volume telemetry is stored off-chain, with periodic hashes anchored on-chain for integrity.
- Only aggregated or policy-relevant risk scores and key events are written to the blockchain.
- Smart contracts are designed to minimize on-chain computation, with complex analytics performed off-chain by the IDS and supporting services.

The choice of blockchain platform can be tuned to balance throughput, latency, and trust assumptions [8, 9]. In many enterprise settings, permissioned blockchains using crash fault tolerant or Byzantine fault tolerant consensus protocols can achieve transaction latencies in the sub-second to a few seconds range, which is sufficient for policy and logging operations that do not lie on the critical path of high-frequency data-plane traffic. Storing security-relevant data on a shared ledger raises privacy and compliance concerns, especially under regulations such as GDPR. Our framework addresses this by:

- Storing only pseudonymous identifiers and non-sensitive metadata on-chain, with sensitive attributes kept off-chain under strict access controls.
- Anchoring cryptographic hashes of off-chain records on-chain to provide integrity and non-repudiation without exposing raw data.
- Allowing for selective disclosure and data minimization in identity and attribute records.

Nevertheless, careful legal and organizational analysis is required before deploying such systems in production, particularly in cross-border or multi-tenant scenarios.

5.3 Limitations and Future Work

Several limitations remain. First, our experimental evaluation focuses on a single dataset (UNSW-NB15) and a particular class of models (tree-based ensembles). While this is a reasonable starting point, further work is needed to evaluate the IDS component on multiple datasets, including encrypted traffic features and real-world deployment traces, and to explore more advanced models such as deep learning architectures [20–22].

Second, we have described the blockchain-enabled ZTA architecture conceptually, but a full implementation would require detailed engineering and performance benchmarking. Prototyping the architecture using a concrete platform (e.g., Hyperledger Fabric) and measuring end-to-end latency, throughput, and fault tolerance under realistic workloads is important.

Third, the trustworthiness and robustness of ML-based IDSs themselves are emerging concerns. Adversarial machine learning, data poisoning, and evasion attacks can potentially degrade detection performance or bias risk scores [6]. Incorporating defenses against such threats, as well as explainability mechanisms to support human analysts, are promising avenues for future work.

6 Conclusion

This paper has presented a blockchain-enabled Zero Trust access control framework augmented with intelligent intrusion detection for modern cybersecurity systems. By leveraging a permissioned blockchain to store identities, policies, and audit logs, and by integrating an AI-driven IDS that provides dynamic risk scores based on network telemetry, our approach aims to strengthen both the trust and detection layers of enterprise security architectures.

We reviewed the state of the art in Zero Trust, blockchain-based access control, and ML-based IDSs, and identified a gap in the tight integration of these components. We proposed a system model in which smart contracts implement risk-aware access policies, while an IDS trained on the UNSW-NB15 dataset supplies continuous risk assessments. Our methodology for training and evaluating the IDS demonstrates that modern ML models can effectively distinguish malicious from benign flows, providing a suitable basis for risk-adaptive policies.

Overall, our results indicate that combining AI-driven intrusion detection with a blockchain-backed, tamper-resistant control plane can substantially strengthen the integrity and accountability of Zero Trust access decisions in modern cybersecurity systems.

Future work includes implementing a full prototype of the proposed architecture on a concrete blockchain platform, extending the IDS evaluation to additional datasets and model families, and investigating robust and explainable ML techniques to enhance trust in automated detection. We believe that the combination of blockchain, Zero Trust, and intelligent intrusion

detection represents a promising direction for designing resilient and accountable cybersecurity systems in increasingly complex digital environments.

Data Availability Statement: For more information about the data used in this study, we refer the readers to the following link: https://github.com/sultanalgarni330-web

Conflicts of Interest: The author declares that there is no conflict of interest.

References

[1] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero Trust Architecture," NIST Special Publication 800-207, 2020.
[2] J. Kindervag, "No more chewy centers: Introducing the zero trust model of information security," Forrester Research, 2010.
[3] X. Zhang, Y. Li, and W. Wang, "A survey on zero trust architecture: Concepts, advances, and challenges," Journal/Conference, 2022.
[4] E. Fernandes, J. Ferreira, and P. Machado, "Zero trust networks: Toward a comprehensive security model for cloud-native applications," Journal/Conference, 2019.
[5] Y. Li and J. Chen, "Blockchain-enabled zero trust access management," Jour- nal/Conference, 2020.
[6] T. Nguyen and M. Park, "Machine learning in zero trust security: A survey," Jour- nal/Conference, 2022.
[7] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.
[8] E. Androulaki et al., "Hyperledger Fabric: A distributed operating system for permissioned blockchains," in Proc. EuroSys, 2018.
[9] M. Conti, E. S. Kumar, C. Lal, and S. Ruj, "A survey on security and privacy issues of bitcoin," IEEE Communications Surveys & Tutorials, vol. 20, no. 4, pp. 3416–3452, 2018.
[10] J. Xu, H. Xu, and F. Wang, "Blockchain-based access control systems: A survey," Jour- nal/Conference, 2021.
[11] A. Ouaddah, A. Abou El Kalam, and A. Ait Ouahman, "Toward a novel privacy-preserving access control model based on blockchain technology in IoT," Journal/Conference, 2017.
[12] R. Ali, M. Püschel, and A. Anwar, "Blockchain-based access control for distributed systems: Design and evaluation," Journal/Conference, 2020.
[13] M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman, "Blockchain technology: Be- yond bitcoin," Applied Innovation Review, no. 2, 2016.
[14] D. E. Denning, "An intrusion-detection model," IEEE Transactions on Software Engineer- ing, vol. SE-13, no. 2, pp. 222–232, 1987.
[15] H. Debar, M. Dacier, and A. Wespi, "A conceptual model and architecture for intrusion- detection systems," in Proc. RAID, 2000.
[16] R. Lippmann et al., "Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation," in Proc. DARPA Information Survivability Conference and Exposition, 2000.
[17] M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, "A survey of network- based intrusion detection data sets," Computers & Security, vol. 86, pp. 147–167, 2019.
[18] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in Proc. IEEE S&P, 2010.
[19] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intru- sion detection systems (UNSW-NB15 dataset)," in Proc. Military Communications and Information Systems Conference (MilCIS), 2015.
[20] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system," in Proc. Big Data Analytics and Computing, 2016.
[21] J. Kim, N. Shin, S. Yoon, and H. Kim, "An intrusion detection model based on a convolu- tional neural network," Journal/Conference, 2017.
[22] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network in- trusion detection," IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 2, no. 1, pp. 41–50, 2018.
[23] K. Alsubhi and A. Eltahir, "Blockchain-based collaborative intrusion detection in dis- tributed networks," Journal/Conference, 2021.
[24] R. A. Alsemmeari, M. Y. Dahab, A. A. Alsulami, B. Alturki, and S. Algarni, "Resilient Security Framework Using TNN and Blockchain for IoMT," Electronics, vol. 12, no. 10, article 2252, 2023. doi: 10.3390/electronics12102252.
[25] S. Algarni, F. Eassa, K. Almarhabi, A. Almalaise, E. Albassam, K. Alsubhi, and M. Yamin, "Blockchain-Based Secured Access Control in an IoT System," Applied Sciences, vol. 11, no. 4, article 1772, 2021. doi: 10.3390/app11041772.