IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Adaptive Behavior - Driven Zero Trust Control Plane for Real-Time Network Security

[1] Poorani S, [2] Nerangen K,

[1] Associate Professor, [2] UG Student

[1,2] Department of Computer Science and Engineering,

[1,2] Sri Venkateswara College of Engineering, Chennai, Tamil Nadu, India.

Doi: https://doi.org/10.56975/ijcrt.v14i7.309051

Abstract: Current computer networks function in cloud computing environments which undergo constant changes because users and devices and services keep connecting to different systems throughout the network. The existing security methods depend on fixed access control rules which determine trust based only on the first authentication and maintain trust until the user session ends. The system creates essential security weaknesses because it allows inside attackers and credential theft and lateral movement to occur when it does not track system operations in real time. The research introduces an Adaptive Behavior-Driven Zero Trust Control Plane which establishes continuous protection for network security operations. The system records all TCP and UDP network data which it converts into flow patterns through specific time intervals that generate network usage data which includes total packets and total bytes and total session time. An unsupervised Isolation Forest model is employed to detect anomalies without requiring labelled data, enabling the identification of unknown and emerging threats. The system uses statistical baselines to assess current trust levels through dynamic analysis of generated anomaly scores in order to determine present system status. The system uses trust levels to control an adaptive policy engine which implements rate limiting and access blocking and real-time alerting functions. The framework functions as a modular system which connects traffic acquisition and anomaly detection and trust evaluation and policy enforcement and visualization to create a cloud-based system that offers real-time monitoring at scale. The testing results show that the system can detect anomalies successfully while applying security measures which boost the security strength and quick response ability of contemporary network protection technologies.

Index Terms - Zero Trust Architecture, Anomaly Detection, Isolation Forest, Network Security, Trust Computation, Real-Time Monitoring, Adaptive Security.

I. INTRODUCTION

The fastest growing areas within computer networks are cloud computing and distributed systems, revolutionizing the way current network environments are built and operate by providing users constant connection and interactivity with applications and services within any network infrastructure. While system evolution has provided greater scalability and flexibility of operations, the nature of distributed systems has created new security challenges because attacks are now directed through the dispersed nature of operations towards a wider attack surface. Current security models relying on perimeter defenses and fixed access controls can only check trustworthiness once during user logon. They grant users access right and then implicitly trust the user during the whole session while failing to monitor the actions that users take during that time. A cause of security threats comes from the fact that the system operates on the

implicit assumption that once the system authenticates a user and gives it credentials to access the network, that user stays in a trusted state. [1] Security threats can no longer be detected by rule-based and signature-driven systems because they rely on a knowledge base of known threats, posing difficulty to situations in which user behavior is always evolving. This means that static checks must be supplemented or entirely replaced with security systems that conduct monitoring constantly and are capable of adapting its security settings to new threats.

Zero Trust Architecture addresses security issues through eliminating all implicit trust and requiring validation for every access request to a system by users. The traditional approach [2] [3] deals with authentication of users in the system based on their identities, while missing out on dynamic analysis of user's behavior. This paper proposes a Zero Trust framework that utilizes the concept of behavior assessment to assign trust score to an entity based on network activity [5], which will in turn govern how the entity is secured. System capture network traffic, analyze at flow-level, employ unsupervised model namely Isolation Forest to detect anomaly and implement different security policies accordingly. It allows a non-stop assessment of trust and therefore leads to a more robust system for future emerging threats[6].

ZERO-DOWNTIME EXECUTION: THE ADAPTIVE ENFORCEMENT PIPELINE

To ensure a modern network security system can detect and respond to threat scenarios without impacting legitimate traffic flow, it uses a zero-downtime execution model. The monitoring, detection, trust establishment and enforcement operations occur simultaneously and stream as a single pipeline. This process fits with the Zero Trust perspective, in that it ensures constant verification and instant enforcement [7] and therefore, allows security decisions to be acted on immediately.

The pipeline continuously monitors the traffic of the network, where each time window of a flow is individually executed and analyzed.[8][9] Unlike traditional systems, which use batch processing or a lag in processing, this mechanism permits a system to instantly act upon any suspicious traffic. The pipeline maintains low-latency and does not suffer from bottlenecks introduced by lightweight feature extraction and an efficient anomaly detection mechanism to delay enforcement logic. For present day AI intrusion detection systems, real-time traffic monitoring approaches are necessary for quick responses required for defense against current threats [10].

CONSTANT PACKET INGESTION AND PROCESSING

The first step in the execution pipeline involves ingesting all network packets from TCP and UDP streams, by utilizing low level network monitoring utilities [11] and directly passing them to the processing pipeline for real-time processing, without any buffering. Parsing the packet allows for retrieval of all necessary metadata such as the IP address and protocol of the source and destination of a packet along with packet size and the timestamp it was recorded at. With this approach of streaming ingestion [12], the continuous transmission of network data [13] is assured and no delays occur due to buffering, maintaining real time monitoring [14][15] mode of operation.

II. PROPOSED SYSTEM ARCHITECTURE

The system processes network traffic and transforms it into security policy using a modular pipeline. By taking raw TCP and UDP traffic as an input, it classifies it into flow-based representation with time-window, which removes the packet layer and provides a session-level abstraction of network traffic flow. Behavioural features, such as the number of packets, the number of bytes, and the session duration, are then extracted to form a structured feature vector.

The feature vector is then passed to an unsupervised Isolation Forest classifier, which distinguishes out of line features from the 'normal' traffic flow (useful in case of new, unidentified attacks as it does not need training on existing labelled data which is impossible if a network is already live. Each flow then produces an output risk score relative to 'normal' data:

The system utilizes a dynamic assessment rather than static thresholds based on std dev or mean values in order to adapt to variations in network behavior; the calculated trust scores are then mapped to the appropriate policy rules of 'High', 'Medium' or 'Low' trustworthiness. Hence the system represents a feedback loop mechanism for continuous monitoring.

WINDOW BASED STREAM PROCESSING

All network packets received are grouped in time intervals known as time windows, each window of a flow generates a representation of the flow at a particular time interval. Each time window represents a single unit of traffic, allowing for:

- Parallel processing of many flows simultaneously.
- Light load and decreased computation demand.
- Constant update of the system's state.

Window-based processing is scalable and provides critical high throughput while maintaining time context.

ISOLATION FOREST -BASED ANOMALY DETECTION

An unsupervised isolation forest classifier is implemented to carry out anomaly detection on network activity. Due to the lack of need for labels unlike in a supervised learning model, novel or emerging threats can be effectively detected through recursively partitioning the feature space and isolating features that differ from the norm. The distance to other features or the number of splits required to separate a feature represent the anomaly value of a data instance. An anomaly score is therefore produced for each window of flows to measure the distance from 'normal' behavior. This classification algorithm has been adopted for its low computational requirements that enable accurate classification while ensuring speed in high-performance network environments that constantly change.

STATISTICAL NORMALIZATION AND SCORE STABILIZATION

It can be expected that anomaly scores will vary based on the general traffic load, and on behavior of the system. To overcome this anomaly scores are normalized over past traffic events so that discrepancies and inconsistencies within anomaly score values are minimized and a measure of deviation from normal behavior can remain constant. Normalization scales anomaly scores in relation to past observations so as to make subtle deviations more prominent while amplifying any anomaly in data from "normal" data. Statistical normalization can also prevent false positive anomalies due to normal traffic variation, and keep sensitivity high toward crucial threats. By standardizing the score scale, trusted computations can be derived reliably.

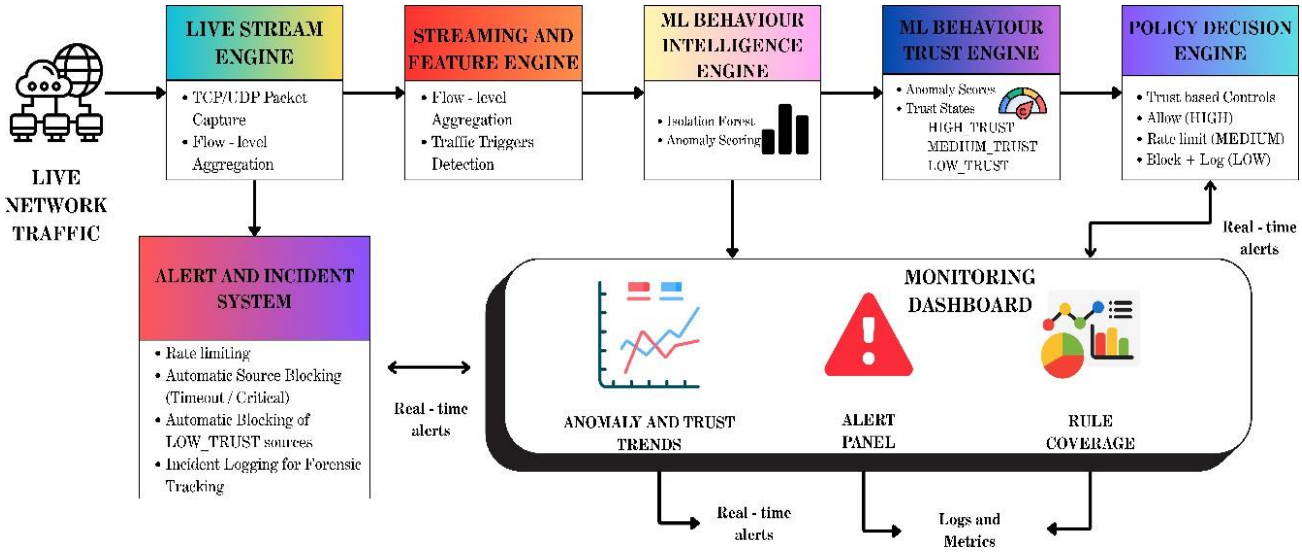

FIGURE 2.1 PROPOSED ARCHITECTURE DIAGRAM FOR ADAPTIVE ZERO TRUST SYSTEM

In Figure 2.1 the anomaly scores are computed for each time window using a trained Isolation Forest model which produces raw anomaly scores. Normalization is then carried out so as to not compare anomalies directly with a fixed threshold but with the dynamic baseline established. The normalization process compensates for dynamic fluctuations in network traffic so that sharp changes don't influence the anomaly scores, and in essence maintain stable behavior for these anomaly scores. The system constantly updates the trustworthiness of a network connection based on the observed behavior, to maintain the current state. This process is in real time and so no anomalies will be missed and threats may be detected much faster.

LOW-LATENCY POLICY ENFORCEMENT

As soon as the trust level of a connection is computed, the enforcement logic immediately follows. Enforcement is a fast and responsive process, since it is directly integrated within the processing pipeline and thus ensures low latency when reacting to threats. The actions taken are based on the trust level:

- High Trustworthy traffic flows are immediately allowed.
- Medium Trust Suspicious traffic is throttled/rate limited.
- Low Trust Malicious traffic is immediately blocked and alert is generated.

These actions are in proportion to the security risk and help minimize false positive alarms while still ensuring safety.

DYNAMIC TRUST COMPUTATION

A trust value is assigned to each flow, based on the calculated anomaly score, compared to a dynamically calculated baseline using the statistics of current and recent observations. This dynamically calculated baseline allows for flexibility in traffic pattern compared to a fixed threshold based on statistical measures of distance. Each individual flow is compared to the dynamic baseline and given a trust state (High, Medium or Low) that is used to drive the policy decision [8]. Hence it has the ability to adapt dynamically compared to making static decisions.

ADAPTIVE POLICY ENFORCEMENT MECHANISM

The determined policy actions are applied in real time to network traffic: they range from traffic shaping to network access control or alerts depending on the determined level of trust in the network traffic. The policy enforcement mechanism's response time is designed to be minimal, allowing security to act quickly

when a threat is detected, and eliminating the long delays that characterized older static systems. Through combining the detection, analysis and enforcement stages, rapid response to malicious traffic patterns and minimum system disruption is enabled.

III. IMPLEMENTATION AND NETWORK EMULATION

The proposed system is realized as a real-time network monitoring and anomaly detection system with a modular architecture. The whole pipeline from traffic acquisition to features extraction, anomaly detection, trust computation, and policy enforcement occurs as continuous data stream. This whole setup is designed in a low-latency manner leading to a real-time estimation and adaptation security policies for cloud - based network environments as to comply with current AI driven IDS trends of behavioral analysis and real time detection.

SYSTEM ENVIRONMENT AND SETUP

The whole system is developed on python-based network monitoring and machine learning tool. Real time packet capture from various interfaces on the network is performed and source and destination TCP and UDP packet are extract from it. The whole system setup is implemented on virtual machine on the cloud which allow us to emulate the real network and a scalable environment to run the experiments under specific conditions of the network.

REAL-TIME TRAFFIC CAPTURE AND PROCESSING

The whole process of continuously collecting network packet and then continuously process without storing it into any buffer. The collected packet are parsed and the source and destination IP address, protocol, packet size and time stamp are extract from it. Data is collected based on fixed time interval to change the packet-based data to flow-based data, thus enabling an effective incremental analysis of the behavior of the network traffic.

FEATURE EXTRACTION AND PREPROCESSING

Based on the derived data from each flow, certain behavior characteristics like number of packet and number of byte and session duration is computed to represent data in a structured format, feature vector, The feature data is normalized based on the variation of characteristics over time in order to develop time-invariant data to ensure the system maintains a stable model which does not vary under changing network conditions.

MODEL INFERENCE AND ANOMALY DETECTION

Unsupervised machine learning model trained for detection of anomaly on flows based on isolation forest algorithm at real time. These anomalies are awarded with anomaly score which is a score for anomaly. The anomaly detection mechanism detect deviation based on trained model which can also be treated as knowledge-based detection of anomalies that can detect unknown threats in real time without use of any labelled data set at a computationally affordable cost for real time system.

TRUST EVALUATION AND DECISION MAKING

Anomaly score is mapped to a trust level by statistical baseline of normal activity which is dynamically update thus making the decision adaptive in nature. Security decisions are based on the trust level by mapping it with pre-defined policy in order to establish correlation between the trust levels and the security decisions as according to the zero-trust model.

POLICY ENFORCEMENT MECHANISM

Based on the obtained trust level the decision taken are implemented on the live traffic such as allow, rate limit or deny by defining respective policies. The real-time integration of security decisions into the policy enforcement mechanism enables a quick response time and thus reducing the impact of potentially malicious traffic on the network.

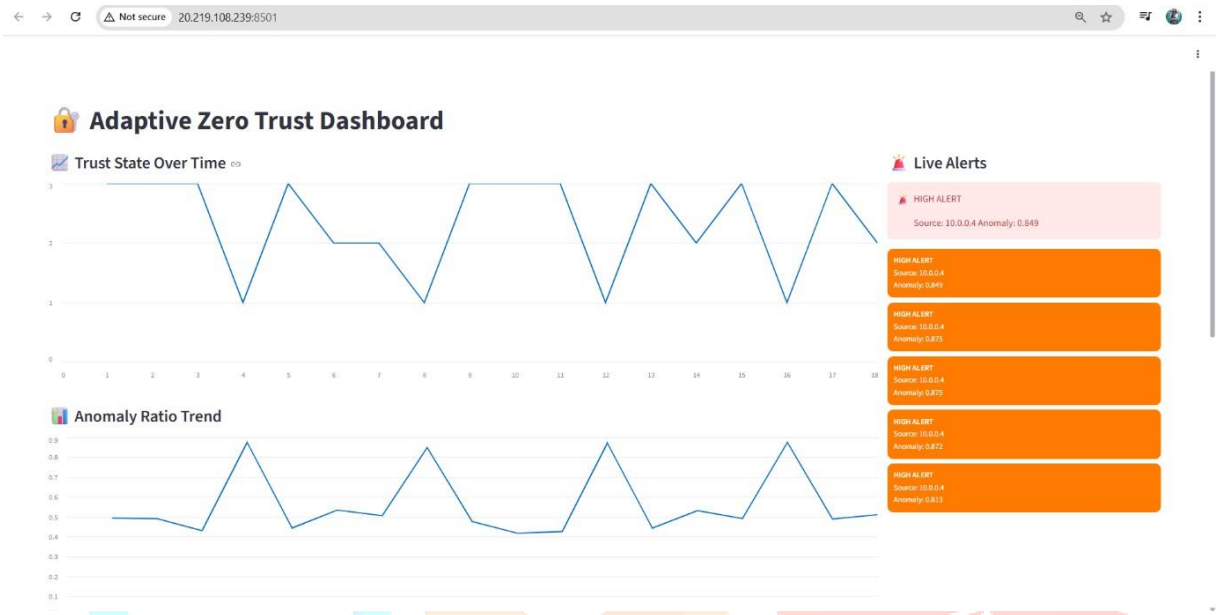

FIGURE 4.1 DASHBOARD VISUALIZATION

Figure 4.1 shows the real time dashboard for displaying anomaly score, trust level and security decision are shown in order to provide an effective analysis to the security personnel. It shows the evaluation metrics computed directly from the system logs, confirming the quantitative performance of the proposed approach.

V. CONCLUSION

In we have demonstrated a behavior driven Zero Trust system with integrated real-time monitoring of traffic, anomaly detection, continuously calculating of trust value and dynamic enforcement of policy. The use of flow - based analysis and an unsupervised Isolation Forest classifier enable our system to adapt and detect anomalous behavior in real-time, using no signatures or labels. Additionally, by constantly evaluating the trust value, we are able to overcome fixed controls and implement security at behavioral level. We have also demons treated that the proposed system can detect and enforce policy on anomalous behavior with minimal latency and it scales up and remains stable in varying traffic load conditions making it a viable solution for cloud - based systems. The fusion of behavior - based analysis with Zero Trust system is an effective security implementation for dynamic environment.

Declarations

Funding: This work was not supported by a specific grant from any funding agency. Public, commercial or not-for-profit sectors.

Conflicts of interests: The authors has no conflict of interest, competing interest.

Data and Code Availability: The implementation of, setup of network traffic simulation and machine learning model parameters used in this study can be provided by the corresponding author upon reasonable request.

REFERENCES

[1] A. M. Abdelmagid and R. Diaz, "Zero Trust Architecture as a risk countermeasure in small- and medium-sized enterprises and advanced technology systems," Risk Analysis, 2025, doi: 10.1111/risa.70026.

[2] H. Alasmary, A. Khormali, A. Anwar, J. Park, J. Choi, A. Abusnaina, D. Nyang, and A. Mohaisen, "Analyzing and detecting emerging Internet of Things malware: A graph-based approach," IEEE Transactions on Network and Service Management, vol. 21, no. 1, pp. 310–324, 2024, doi: 10.1109/TNSM.2023.3324109.

[3] S. Ahmadi, "Autonomous identity-based threat segmentation for zero trust architecture," Cyber Security and Applications, vol. 3, art. no. 100106, Dec. 2025, doi: 10.1016/j.csa.2025.100106.

[4] S. Axelsson, "Intrusion Detection Systems: A Survey and Taxonomy," Technical Report, Chalmers University of Technology, 2000.

[5] E. Bertino and N. Islam, "Botnets and Internet of Things security," Computer, vol. 50, no. 2, pp. 76–79, 2017, doi: 10.1109/MC.2017.62.

[6] A. R. Bilipelli, "AI-driven intrusion detection methods for large-scale cybersecurity networks," Internet of Things, vol. 27, art. no. 101227, 2024, doi: 10.1016/j.iot.2024.101227.

[7] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.

[8] M. Conti, A. Dehghantanha, K. Franke, and S. Watson, "Internet of Things security and forensics: Challenges and opportunities," Future Generation Computer Systems, vol. 131, pp. 266–277, 2023, doi: 10.1016/j.future.2022.12.014.

[9] J. Kindervag, "Build Security Into Your Network's DNA: The Zero Trust Network Architecture," Forrester Research, 2010.

[10] A. A. Laghari, A. A. Khan, A. Ksibi, F. Hajjej, N. Kryvinska, A. Almadhor, M. A. Mohamed, and S. Al-Subai, "A novel and secure artificial intelligence-enabled zero trust intrusion detection in industrial Internet of Things architecture," Scientific Reports, vol. 15, no. 1, p. 26843, 2025, doi: 10.1038/s41598-025-11738-9.

[11] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation Forest," in Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, 2008, pp. 413–422, doi: 10.1109/ICDM.2008.17.

[12] National Institute of Standards and Technology, "Security and Privacy Controls for Information Systems and Organizations," NIST Special Publication 800-53, Rev. 5, Gaithersburg, MD, USA, 2020, doi: 10.6028/NIST.SP.800-53r5.

[13] National Institute of Standards and Technology, "Zero Trust Architecture," NIST Special Publication 800-207, 2023, doi: 10.6028/NIST.SP.800-207.

[14] A. Paya and A. Gómez, "Enhancing software-defined perimeters with integrated identity solutions and threat detection for robust zero trust security," International Journal of Information Security, vol. 24, no. 4, pp. 1–20, 2025, doi: 10.1007/s10207-025-01099-9.

[15] S. Poorani and R. Anitha, "Privacy-Preserving Cloud Data Security: Integrating the Novel Opacus Encryption and Blockchain Key Management," KSII Transactions on Internet and Information Systems, vol. 17, no. 11, pp. 3182–3203, 2023, doi: 10.3837/tiis.2023.11.015.