electronics

MDPI

Article

A Blockchained AutoML Network Traffic Analyzer to Industrial Cyber Defense and Protection

Alexandros Papanikolaou [1], Aggelos Alevizopoulos [1], Christos Ilioudis [2], Konstantinos Demertzis [3,*] and Konstantinos Rantos [3]

[1] Innovative Secure Technologies P.C., 60 Monastiriou St., 54627 Thessaloniki, Greece
[2] Department of Information and Electronic Engineering, International Hellenic University, 57400 Thessaloniki, Greece
[3] Department of Computer Science, International Hellenic University, 65404 Kavala City, Greece
* Correspondence: kdemertzis@teiemt.gr

Abstract: Network traffic analysis can raise privacy concerns due to its ability to reveal sensitive information about individuals and organizations. This paper proposes a privacy-preserving Blockchained AutoML Network Traffic Analyzer (BANTA). The system securely stores network traffic logs in a decentralized manner, providing transparency and security. Differential privacy algorithms protect sensitive information in the network flow logs while allowing administrators to analyze network traffic without the risk of leakages. The BANTA uses blockchain technology, where smart contracts automate the process of network traffic analysis, and a multi-signature system ensures the system's security, safety, and reliability. The proposed approach was evaluated using a real-world network traffic dataset. The results demonstrate the system's high accuracy and real-time anomaly detection capabilities, which makes it well-suited for scalable cybersecurity operations. The system's privacy protection, decentralized storage, automation, multi-signature system, and real-world effectiveness ensure that the organization's data is private, secure, and effectively protected from cyber threats, which are the most vexing issue of modern cyber-physical systems.

Keywords: cyber threat intelligent; cyber threat information; network traffic analysis; industrial environment; blockchain; differential privacy

Citation: Papanikolaou, A.; Alevizopoulos, A.; Ilioudis, C.; Demertzis, K.; Rantos, K. A Blockchained AutoML Network Traffic Analyzer to Industrial Cyber Defense and Protection. *Electronics* **2023**, *12*, 1484. https://doi.org/10.3390/electronics12061484

Academic Editor: Tuan-Vinh Le

Received: 11 February 2023
Revised: 10 March 2023
Accepted: 20 March 2023
Published: 21 March 2023

1. Introduction

The new cyberspace is shaped by a highly interconnected digital environment, which provides new possibilities and opportunities for enterprises to create extroversion activities and behaviors. However, this new cyber-ecosystem faces several concerns, including cybercrime, advanced persistent threats, and zero-day attacks. These sophisticated threats circumvent traditional defense strategies and require comprehensive control over all attempts to exploit the system's weaknesses [1].

Using state-of-the-art cryptography in malware code, combined with the Blind Proxy Redirection (BPR) method, makes identifying Command and Control (C&C) servers extremely difficult. For example, malware and botnets constantly seek ways to cover their identity or evade the detection from the Intrusion Detection/Protection System (IDS/IPS) [2]. The communications are made via secret dynamic Domain Name System (DNS) services using hundreds of random IP addresses. The most advanced malware utilizes the Tor network's chaotic nature to encrypt botnet traces and change attack vectors. For example, Tor uses port 443, and the resulting traffic is identical to real HTTPS traffic. The most efficient Tor-based attack prevention approach and investigation of malware communications is the statistical examination of variations in the Secure Sockets Layer (SSL) protocol. Also, Indicators of Compromise (IOCs), such as Malware signature IDs, malicious IP addresses, malicious checksum MD5, suspicious URLs, or domain names of

Botnets, can help security decision-making. But network traffic analysis is one of the most efficient methods to recognize and identify malware vulnerabilities [3,4].

The following are the essential methods in which network traffic analysis can aid in the timely identification and reaction to cyberattacks [5,6]:

1. Identification of Anomalies: Network traffic analysis can help identify anomalies or unusual patterns in network traffic that may indicate a cyberattack. For example, a sudden increase in traffic from a specific IP address or an unusual rise in the data transfer volume can be signs of a cyberattack.
2. Real-time Monitoring: Network traffic analysis can provide real-time monitoring of network traffic, allowing organizations to detect and respond to cyberattacks as they happen quickly.
3. Threat Detection: Network traffic analysis can be used to detect known threats and vulnerabilities, such as malware, viruses, and hacking attempts. This information can be used to implement appropriate security measures to prevent or mitigate these threats.
4. Improved Incident Response: Network traffic analysis can provide valuable information that can be used to improve an organization's incident response capabilities. For example, by analyzing network traffic patterns and data, organizations can identify the sources of a cyberattack and the methods used to compromise their systems.
5. Better Understanding of Attack Techniques: Network traffic analysis can also provide insight into the techniques used by attackers, allowing organizations to understand better the methods used to penetrate their systems and to develop more effective countermeasures.

By providing real-time monitoring and threat detection, network traffic analysis can help organizations quickly identify and respond to security incidents, thereby reducing the impact of a cyberattack [7]. This study proposes BANTA, an innovative solution that offers security and transparency by decentralized storing of network traffic logs securely. It is a privacy-preserving block-chained autoML network traffic analyzer that securely stores network traffic logs in a decentralized manner aims privacy preserving, providing transparency and security. Differential privacy algorithms protect sensitive information in the network flow logs while al-lowing administrators to analyze network traffic without the risk of leakages. The BANTA uses blockchain technology, where smart contracts automate the process of network traffic analysis, and a multi-signature system ensures the system's security, safety, and reliability. Differential privacy algorithms shield private data from leaks, enabling the examination of network traffic without the fear of security breaches.

Using blockchain technology, the BANTA's multi-signature system protects the computing system's security, and dependability while smart contracts automate network traffic analysis. The objective of this study is to apply a novel approach for detecting advanced persistent anomalies using privacy-preserving auto-ML techniques to network traffic analysis. The proposed approach aims to overcome the limitations of existing anomaly detection methods which are often time-consuming, resource-intensive, and unsuitable for real-time detection. Our main contribution is a system that employs next-generation learning algorithms to analyze network traffic data and identify patterns that indicate the presence of rare cybersecurity events.

This paper's remaining sections are structured as follows: Section 2 provides a literature review, motivation, and novelty of the study methodology. The architecture of CTI2SA is described in Section 3. Section 4 presents the proposed blockchained network traffic analyzer. Section 5 presents a use-case scenario of its implementation, in Section 6 we discuss the results of our work, and finally, Section 7 presents the conclusions, limitations, and future research.

2. Related Work

We investigated the recent literature on decentralized network traffic analysis, anomaly detection, privacy protection, and multi-signature systems to identify gaps in existing knowledge and make informed decisions about designing and implementing the proposed system. For example, Lim and Stadler [8] discovered that as the complexity and variety of

networks expand, the capacity to produce views of a network in a short amount of time becomes increasingly significant. These views incorporate information from numerous remote network points and can help an administrator better understand the interdependencies and interactions between network parts and traffic circumstances. Performance monitoring and fault management are examples of applications that could benefit from such "near real-time" network views. Their study describes a distributed management infrastructure design that enables the computation of such statements. The authors' architecture employs a database strategy that blends the expressive capability of SQL with distributed algorithms. The authors discuss the system's implementation on a platform of embedded Linux devices connected to a network of routers. In addition, they demonstrate how the system may be utilized as a robust distributed real-time monitoring platform. Finally, they derive a performance model for the design and validate it through trials. One constraint of the system is the semantics of Weaver Query Language (WQL)—"join" procedures are restricted to tables on the same Weaver Active Nodes (WAN). Their architecture mandated this because linking tables on separate WANs can be inefficient, necessitating the transport of full tables across WANs. The syntax of WQL reflects this limitation by prohibiting the explicit addressing of individual nodes. On the other hand, there is no mechanism for privacy protection.

The authors of [9] suggest a method for formalizing the normal and abnormal behavior of the system in a set of useful features, and criteria are developed that enable the detection and identification of various forms of network anomalies in order to address the issue of looking for anomalies in networks. The study explores statistically based techniques for anomaly detection, including fractal traffic analysis. The challenges of detecting network attacks are examined, and similar statistical characteristics are shown in the variance and mean change. The sample means, sample variance, entropy, and anti-kurtosis all experienced a dramatic surge during the abnormality, according to the analysis of changes in statistical traffic characteristics. The collected data demonstrate that problems and symptoms can be found using the indicators above in intrusion detection tasks.

Network coding alone is insufficient to eliminate privacy risks in multi-hop wireless networks. This research [10] provides a random network coding technique with the Blowfish encryption algorithm to ensure source anonymity and conceal message data. The proposed algorithm incorporates arbitrary encoding that extends aspects of homomorphic encryption, such as Paillier encryption, utilized in existing systems to prevent traffic analysis attacks successfully. Each wireless user exploits the property of inverting explicit encoding vectors to retrieve source packets with high probability. The proposed system's strength and efficacy are visible through theoretical analysis and simulation data evaluation. The proposed system has unique privacy-preserving features like packet flow intractability, which can resourcefully stop the traffic analysis attacks such as flow locating and message content confidentiality. Also, the system is faster and more competent than traditional privacy preservation schemes but needs to be more credible for the scaling abilities, flexibility, complexity, and resource requirements.

In order to face the above challenge, this study [11] offers a lightweight dual privacy-preserving navigation system for automotive networks by inventing a novel signature technique and combining cryptographic primitives to maintain double privacy, including personally identifiable information and route-related information of intelligent vehicles. The correctness analysis of the suggested system, the security proof of the intended signature scheme utilized in the proposed technique, and the privacy analysis of the proposed approach are described in detail. In addition, the performance of the suggested system is analyzed and compared to existing methods to demonstrate the effectiveness of the proposed method. Focusing on whether the system can combine lightweight zero-knowledge proof technology or differential privacy to provide excellent personal information concealment and efficiency enhancement is crucial to the system's improvement.

Although data-driven systems like efficient routing necessitate precise real-time data, they are vulnerable to data integrity attacks. Thus, we suggest a multi-tiered anomaly

detection system that uses the distributed RSU network's unused processing power in addition to the cloud for quick, real-time detection. The authors introduce a novel real-time anomaly detection methodology in this paper [12]. They also provide a constrained clustering technique for RSU placement throughout the network, focusing on applying our framework in smart-city transportation systems. The suggested methods greatly reduce processing requirements while keeping performance comparable to current state-of-the-art anomaly detection systems, as shown by extensive experimental validation using traffic data from Nashville, Tennessee.

Recently, a decentralized nuclear-norm minimization-based algorithm was developed to solve the low-rank matrix completion problem in a mesh network. In this paper, we consider a two-tier network and propose a new decentralized approach based on the Riemannian optimization. These authors [13] proposed clustering and consensus sharing method achieves a balance between the performance guarantee and the computational cost: the proposed method distributes the computational burden over the agent nodes, while exhibits a recovery performance close to its centralized counterpart. In addition, their Riemannian optimization-based approach scales well with the size of the problem, hence it is more favoured for handling large data sets in IoT than the nuclear-norm minimization-based algorithm. Numerical simulations are performed to demonstrate the effectiveness of the proposed approach, which is able to solve problems of size 2000×2000 of rank 5 with 50 agent nodes in 10 s.

This paper [14] suggests a dependable and effective solution for traffic monitoring that incorporates blockchain and the Internet of Vehicles technologies. The proposed system ensures dependability, efficacy, and security during trade execution. This application's expansion requires various enhancements. The authors first create a lightweight blockchain-based design for information exchange to model the interactions between traffic administration and automobiles. Second, they define the utility functions for the entities in this system and create a budgetary auction mechanism that encourages cars to engage in collection duties actively. Their algorithm ensures that the total payment for the selected automobiles does not exceed a predetermined budget and maintains the integrity of the auction process by preventing certain vehicles from proposing inflated bids for superior utilities. The solution employs contemporary encryption techniques and digital signature technologies to safeguard communication contents against privacy leaks. The most problematic aspect of the suggested strategy, which is evaluated for its reliability, is that the authors undertake numerical simulations to assess the dependability of the trading framework and the performance of their algorithms only on an experimental scale rather than in a real-world setting.

Also, this article [15] devised and executed a blockchain-based multi-signature solution to digitally alter supply chain governance in multi-tier food chains, mainly geographically distributed food supply chains. Utilizing an experimental case study, the design, implementation, and evaluation of a blockchain-based multi-signature solution deployed on the Smart Trade Networks (STN) Proof of Authority (PoA) blockchain system for data collection and validation in a beef supply chain context was demonstrated. The multi-signature technique was developed using a case to trace 92 cattle and meat product shipments from farm to food service through critical events. The use-case deployment on the STN PoA blockchain technology records approximately 6000 data points. The real-world deployment demonstrates the capability of the blockchain-based multi-signature approach to digitally improve beef supply chain governance by enabling whole-of-chain transparency and trustworthy information sharing. It also helps supply chain professionals comprehend how to unlock blockchain's supply chain transformation potential. This exploratory study has some limitations, despite demonstrating the viability of the blockchain-based multi-signature method. Due to existing low-tech capabilities across the supply chain, the data were collected from conventional sources and uploaded to our system using digital photographs and Excel spreadsheets. Some data elements, such as location, should ideally be sent automatically to our blockchain system using IoT and tagging technologies.

The above research studies have focused on generic methods and systems that do not meet the requirements of modern information systems. As a result, these studies do not considerably contribute to greater awareness and understanding of the risks associated with network traffic flow systems and the severity of attacks against them, which typically result in significant damage and financial losses.

The smart contact layer is a relatively more technical and considerably new layer in the blockchain. It became famous in the second era of blockchain, named blockchain 2.0 when the Ethereum platform provided users the functionality of developing applications. Many contracts in a well-established blockchain network are pre-developed and usually contain bugs. The malicious participants in the network always try to find out loopholes by any means and smart contracts are their recent targets because it's easy for a non-technical person to identify bugs and honeypots in smart contracts. From the perspective of detection of anomalies in this specific layer, the woks are divided into multiple types ranging from the identification of contracts restricting dependent transactions to highlighting faulty signals being transmitted via the deployment of a smart contract. However, the most famous works in anomaly detection over this layer have been carried out from the perspective of detection of Ponzi schemes [16] and critical security threats in the contracts, such as hacking.

Trying to fill the gaps in the literature, the proposed work differs significantly from existing systems in terms of the philosophy of protection of modern information systems. Specifically, the proposed BANTA is a holistic approach that combines the real needs of modern information systems and is novel in several ways. Firstly, it combines multi-signature design and differential privacy algorithms to provide a secure and privacy-preserving solution for network traffic analysis, filling the existing literature gap. Secondly, it uses blockchain technology to store network traffic logs in a decentralized manner, providing a secure and transparent platform for network traffic analysis. Thirdly, it implements intelligent contracts to automate the process of network traffic analysis, providing a more efficient and streamlined approach compared to traditional network traffic analysis techniques. Fourthly, auto-machine learning algorithms can automatically learn from data without requiring manual feature engineering or model selection by experts. This reduces the time and effort required for network traffic analysis and makes it possible for non-experts to perform the analysis. Fifthly, auto-machine learning algorithms automatically adapt to changes in network traffic patterns without requiring manual updates to the models. This increases the accuracy and reliability of the analysis over time and reduces the risk of human error. Sixthly, auto-machine learning algorithms provide a more accurate and granular analysis of network traffic patterns than traditional rule-based methods. Specifically, auto-machine learning algorithms automatically identify complex relationships and anomalies in the network traffic data, which is impossible with conventional rule-based methods. Finally, auto-machine learning algorithms implement differentially private algorithms to protect sensitive information in the network traffic logs while providing meaningful analysis results.

In conclusion, this paper makes a valuable contribution to the field of network traffic analysis by proposing a secure and privacy-preserving solution that combines a multi-signature system and differential privacy algorithms. The proposed method provides a robust and trustworthy platform, addressing the security and privacy challenges faced by traditional network traffic analysis techniques. In addition, the use of auto machine learning methods provides several novelties to the field, including improved accuracy and reliability of the study, reduced time and effort required for research, automatic adaptation to changes in the network traffic patterns, and privacy-preserving analysis of sensitive information. These novelties make the BANTA approach a promising solution for distributed network traffic analysis in various scenarios.

3. CTI2SA Architecture of the Cyber-Pi Project

The suggested CTI2SA architecture [17] is an adaptable security solution that integrates different control methods and digital security technologies for modern computing systems and networks. It provides a centralized location for analysis, alert, compliance, and reporting in response to the evolving organizational structures of a modern, multidimensional enterprise. It includes several sophisticated mechanisms for monitoring data integrity, notifying new risks, identifying and recording security occurrences, and promptly responding to automated operations. The system focuses initially on the timely identification of events through automated, detailed log analysis, and updating and enhancing the predictability of the system is based on collecting cyber-threat data from trusted open-access sources [18]. The proposed architecture uses component mapping to adapt to the organization's business operations and information systems requirements.

Cyber-cognition in the STIX 2. x standard is adapted based on comparing cyber threats with the organization's characteristics. The privacy policy production subsystem includes methods for modulating and exporting SIGMA rules. To offer a high level of cyber security on supported systems and applications, the proposed solution uses a data-driven Network Traffic Analyzer. The visualization and interface subsystem is aided in each phase of its operation, and the design of preventative countermeasures is limited to identifying specific dangers that may harm the organization and the establishment of general priorities. This sophisticated feature develops processes for intelligent decision-making or adaptability that assure seamless operation. Figure 1 provides a basic overview of CTI2SA components and functionalities.

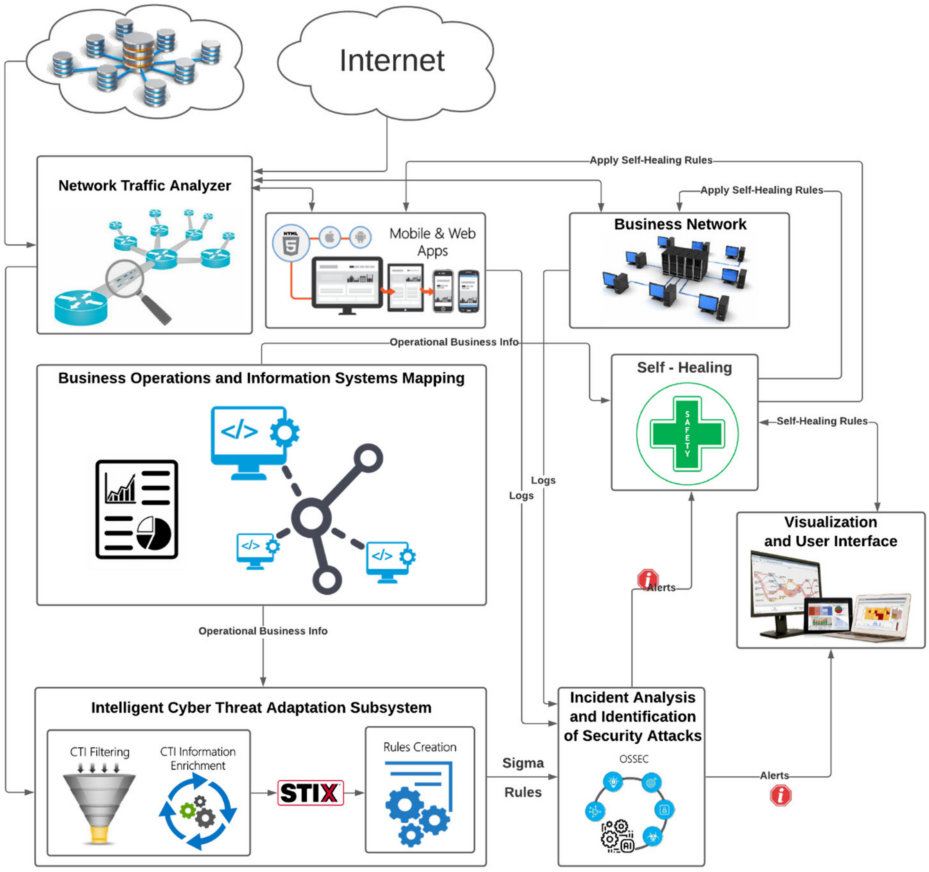

Figure 1. The Cyber Threat Intelligent Information Sharing Architecture (CTI2SA).

The subsystems and mechanisms constituting the CTI2SA architecture are presented in detail below [17]:

1. OSSEC HIDS. The OSSEC Host-based Intrusion Detection System evaluates the integrity of supervised information infrastructure files at the application and system levels using threat detection techniques based on signatures and statistical abnormalities. It can be configured to collect events from devices where agents are impractical and contain rules for monitoring and assessing specialized security events.
2. Decoders. They analyze the logs of the target environment using default and custom decoders with settings that are compared to the log content to detect events. The incident notifications under consideration are generated based on accessible rules that implement various security policies and route all incoming correspondence for control.
3. Intelligent use of CTI. The Intelligent Cyber Threat Adaptation Subsystem collects and analyses data from Cyber Threat Information (CTI) Sources, compares the results to those of the supervised Information System, and records the threats to the target network and information infrastructure. The intelligent aggregation mechanism is routinely updated with IOCs, and RSS feeds gathered from various dependable sources, like Malware Information Sharing repositories and Threat Intelligence Sharing Platforms. The filtering process preserves relevant data and compares it to the specific information infrastructure. Custom knowledge is generated in STIX 2.x files to draft SIGMA rules.
4. Business operations and information systems mapping. The Dependency Mapper utility implements the mechanism's fundamental procedures, and the adaption of the depmapper to the mapping method enhances the exporting of graphs to image files and data interchange capabilities. The mapping process explains the organization's qualities, while the technical environment is displayed on the system's console. The layout model depicts the components, hardware components (nodes), and their interconnections. The descriptions include details on the object of operation, available software services, geographic location, structural and procedural dependencies, data dependencies, and risk level. The importance or sensitivity of the accessible data and services decides the risk. An analyzer collects the data and enters it in JSON format for usage by other subsystems.
5. Visualization and user interface. Visualization of security data received from various log sources involves creating diagrams, graphs, and other visual content. Categorizing alerts is beneficial and permits the development of a monitoring strategy. The user interface mechanism allows for the approval or activation of self-healing system procedures and enables the rapid capture of vital data and prompt response to events. It also supports web environments, enabling mobile devices and web browsers to access services.
6. Self-Healing Policies. Self-Healing Policies are derived from the organization's decision analysis and prioritizing procedures and are documented in the system database in a transparent and interoperable manner.
7. ANTA. It is a data-driven module designed to manage and classify network traffic. Based on the Lambda architecture, it improves active cybersecurity approaches related to traffic analysis by merging batch and stream processing to manage enormous volumes of data. It includes an auto-model-selection method and selects the ML model with the most outstanding performance among competitors to maintain the efficacy of the architecture's threat identification capabilities. It is a flexible self-adapting system that automates malicious traffic detection, generates alerts for further checks, and implements appropriate security policies to minimize the organization's attack surface.

As mentioned before, a primary component of the CTI2SA architecture is the data-driven AutoML Network Traffic Analyzer (ANTA), which offers a high level of cyber security. But network traffic analysis can raise privacy concerns because it can reveal sensitive information about individuals and organizations. Some of the most critical privacy issues are [10,19,20]:

1. Personal Information Leakage: Network traffic analysis can uncover sensitive personal information, such as financial information, health records, and social security numbers. This information can be used for malicious purposes, such as identity theft and fraud.
2. Unauthorized Surveillance: Network traffic analysis can also monitor and track an individual's online activities, including their communication and browsing habits. This can result in the violation of an individual's privacy and civil liberties.
3. Discrimination: Network traffic analysis can also be used to profile individuals based on their online activities, leading to discrimination based on race, gender, sexual orientation, or political views.
4. Misuse of Data: The data collected through network traffic analysis can be misused by individuals or organizations with malicious intent. This can include using the data for unauthorized marketing or advertising purposes.
5. Inadequate Security Measures: In some cases, the security measures put in place to protect the data collected through network traffic analysis may need to be improved, making it vulnerable to hacking and other cybercrime.

It is essential for individuals and organizations to be aware of these privacy issues and to take steps to protect their sensitive information from being compromised through network traffic analysis [21].

To address these challenges, this paper proposes a blockchain-based system that combines a multi-signature system and differential privacy algorithms to provide a secure and privacy-preserving solution for network traffic analysis. The proposed system uses smart contracts to automate the process of network traffic analysis and store network traffic logs in a decentralized manner, providing transparency and security. The multi-signature system ensures that a specific set of parties must sign off on any changes to the smart contract's behavior or data, preventing rogue administrators from altering the smart contract in ways that could compromise its security or reliability. The differential privacy algorithms protect sensitive information in the network traffic logs while allowing network administrators to perform network traffic analysis.

4. The Proposed BANTA

The data-driven ANTA system manages and categorizes network traffic as mentioned above. It is a versatile self-adapting system that automates the detection of harmful traffic, produces alarms for additional inspections, and applies applicable security rules to reduce the organization's attack surface. Combining batch and stream processing to handle massive amounts of data, this Lambda architecture-based solution enhances dynamic cybersecurity tactics connected to traffic analysis. To retain the effectiveness of the architecture's capability for threat identification, it incorporates an auto-model-selection approach that chooses the ML model with the most exceptional performance among rivals. At the same time, the proposed BANTA improves the ANTA's architecture by incorporating blockchain technology and privacy-preserving features. To be more precise, BANTA is a blockchain-based system that combines a multi-signature system and differential privacy algorithms to offer a secure and privacy preserving solution for network traffic analysis.

A graphical depiction of the Lambda architecture, which is the main core of the ANTA and BANTA architectures, is shown in the following Figure 2.

The blockchain architecture [22] is a distributed database or global registry that maintains logs of all network transactions. Transactions are combined into blocks sorted based on a cryptographic hash. An open public-private key pair is formed for each user linked to the corresponding account. The blocks are added to the blockchain at regular intervals, creating a linear sequence where each block states the previous block's hash. Each blockchain user has access to the entire transaction log and can check the hash of each new block.

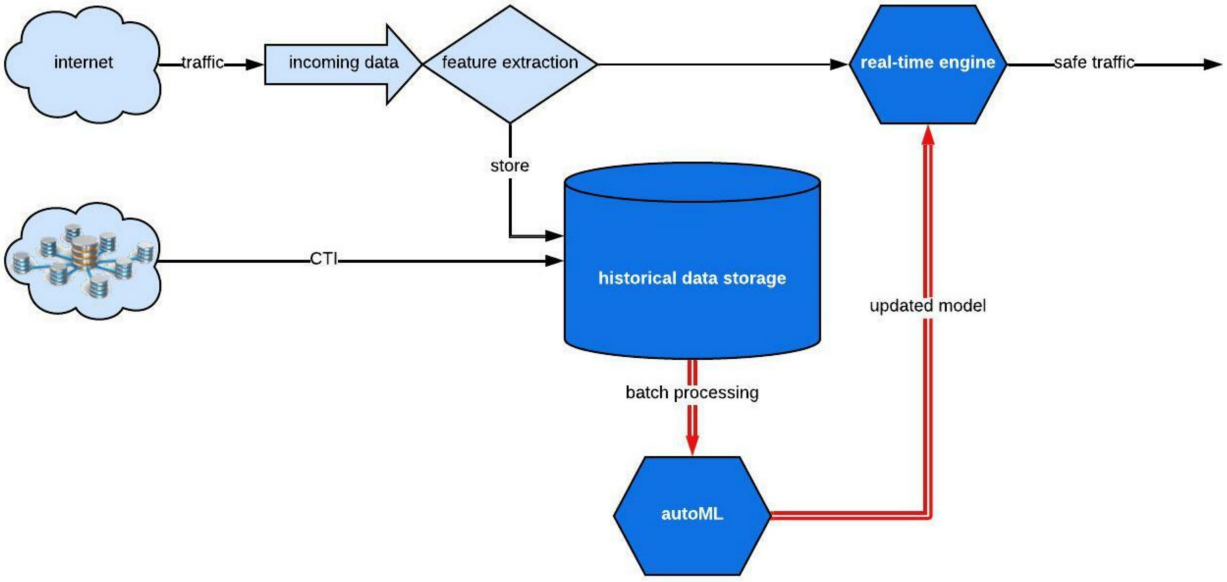

Figure 2. The proposed Architecture (the main core of ANTA and BANTA systems) which is based on the Lambda architecture.

The blockchain is ideal for managing industrial assets' interests as it achieves transaction confidentiality and selective access between authorized participants. It is based on the Byzantine Fault Tolerant (BFT) consent algorithm and is jointly controlled by the network's members.

Blockchain technology used in the proposed Network Traffic Analyzer in the following ways:

1. Decentralized Network Traffic Logging: The blockchain store network traffic logs in a decentralized manner. This increases network traffic data's security and transparency, as records cannot be easily altered or deleted.
2. Verification of Traffic Data: The immutable nature of blockchain records used to verify the accuracy of network traffic data. This help to ensure that network administrators and security personnel have accurate information to work with.
3. Secure Sharing of Traffic Data: Blockchain technology securely share network traffic data between multiple organizations or stakeholders. This help to improve collaboration and information sharing in network security investigations.
4. Automated Network Traffic Analysis: Smart contracts on the blockchain used to automate network traffic analysis. Specifically, a smart contract automatically triggers an alert or action when specific network traffic patterns are detected.
5. Compliance and Auditing: The blockchain store network traffic analysis records for compliance and auditing purposes. This help demonstrates that an organization complies with relevant regulations and standards. Each component/actor has an identity and policies that define access control and governance. Peers are a fundamental element of the web, and they have an identity of their own and are managed by the administrator of an organization. Transport Layer Security (TLS) is used to secure communication between nodes, and cryptographic operations are delegated to a Hardware Security Module (HSM). This allows peers to endorse transactions and ordered nodes without exposing their private keys.

To ensure that network administrators and security personnel have accurate information to work with, the BANTA improved in the following ways:

1. Add access control: Added access control to the smart contract to ensure that only authorized parties can access network traffic logs and analysis results. Specifically, the BANTA uses the identity management system uPort to manage access to the smart contract, allowing only specific addresses to read the logs and trigger alerts.
2. Used reliable data sources: To ensure that network traffic data is accurate and trustworthy, the proposed system uses reliable data sources. Specifically, the smart contract

receives network traffic data directly from recognized network switches, routers, or a trusted third-party data source.

3. Add data validation: To further ensure the accuracy of network traffic data, was add validation logic to the smart contract. Specifically, checks were included to ensure that traffic data is well-formed and contains all required information or that the system uses cryptographic signatures to validate the authenticity of the data.
4. Use multi-sig: The BANTA implemented a multi-signature system to ensure that a specific set of parties must sign off on any changes to the smart contract's behavior or data. This prevents rogue administrators from altering the intelligent contract in ways that could compromise its security or reliability.

The architecture of the system is designed to provide a comprehensive solution for network traffic analysis, with features for data storage, encryption, decryption, signature verification, alerts and notifications, and reporting. Specifically, the system architecture is divided into the following components [15,23,24]:

1. Smart Contract: This is the system's main component and is responsible for managing and storing network traffic logs securely and transparently.
2. Logs Mapping: A mapping data structure that stores the encrypted noisy network traffic logs, where the log ID is used as the key, and the encrypted noisy log value is used as the value.
3. Administrators Array: An array that stores the addresses of authorized administrators who are authorized to access and modify the network traffic logs.
4. Threshold Variable: A variable that stores the multi-signature threshold, which ensures that a specific set of parties must sign off on any changes to the smart contract's behavior or data.
5. Epsilon Variable: A variable that stores the differential privacy noise used to ensure the privacy of the network traffic logs.
6. Add/Remove Administrator Functions: Functions that allow the contract owner to add or remove authorized administrators.
7. Update Threshold/Epsilon Functions: Functions that allow the contract owner to update the multi-signature threshold and the differential privacy noise.
8. Store Log Function: A function responsible for storing network traffic logs in the blockchain. This function encrypts the records using Elliptic Curve Digital Signature Algorithm (ECDSA) to ensure their privacy and security.
9. Privacy-preserving method: The Secret Sharing Protocol (SSP) is used as a privacy-preserving method for sharing secrets among multiple parties. The secret is split into shares, each distributed to a different party. The key advantage of the secret sharing protocol is that no single share reveals any information about the original secret, and the secret can only be reconstructed by combining a minimum number of shares. Moreover, the secret-sharing protocol is used to share secrets while preserving the participants' privacy. For example, in a secure multi-party computation scenario, each participant can share their private inputs using a secret sharing protocol without revealing them to other participants. This ensures that each participant's input remains private while allowing for the computation of the desired function on the combined inputs.
10. Differential Privacy method: Zero-knowledge proofs (ZKP) is the differential privacy method that allows one party to prove to another party that they know a specific piece of information without revealing any information about that information itself. The proof is designed to be convincing enough to the verifier but without revealing any information that can be used to reconstruct the secret.
11. Event Emitters: The intelligent contract emits events whenever a new network traffic log is stored or whenever the multi-signature threshold or differential privacy noise is updated. Network administrators and security personnel can monitor these events to track changes and updates to the system.

12. Signature Verification: The intelligent contract includes functions for verifying authorized administrators' signatures before executing any system changes. This ensures that only the specified parties can access and modify the network traffic logs.
13. Decryption: The intelligent contract includes functions for decrypting the network traffic logs for authorized administrators. This allows administrators to access the raw records for analysis and troubleshooting.
14. Security Measures: The intelligent contract includes various security measures to ensure its integrity and protect against potential attacks. These measures include secure storage of sensitive information, such as private keys, and using specific cryptographic algorithms for encryption and decryption.
15. Blockchain Network: The smart contract is deployed and runs on a decentralized blockchain network like Ethereum. This allows for secure, transparent, and tamper-proof storage of network traffic logs and ensures that the records cannot be easily altered or deleted.
16. Client-side Applications: The system can be accessed and managed through client-side applications that interact with the smart contract. Administrators can use these applications to store network traffic logs, view, and decrypt logs, and execute changes to the system, such as updating the multi-signature threshold or differential privacy noise.
17. Data Analytics Tools: The system can be integrated with various data analytics tools for network traffic analysis. These tools can analyze the network traffic logs stored in the blockchain and detect patterns or anomalies that could indicate potential security threats.
18. Alerts and Notifications: The system can trigger alerts or notifications based on specific network traffic patterns. These alerts can be used to notify administrators or security personnel of potential security threats and to take appropriate action.
19. Auditing and Reporting: The system can be configured to provide auditing and reporting capabilities, allowing administrators to track changes and updates to the design and view the history of network traffic logs stored in the blockchain.

This system architecture provides a secure and transparent way for network administrators and security personnel to monitor and analyze network traffic data while protecting individual data points' privacy.

The critical element of the proposed architecture is the smart contract. A smart contract is a blockchain-based program that executes when specific criteria are met. Typically, it is used to automate the implementation of an agreement so that both parties may be sure of the conclusion without the need for an intermediary or any delay [25].

The proposed smart contract aims to provide a more secure and transparent solution for network traffic analysis, as network traffic logs are stored decentralized, and only authorized parties can access the data and trigger alerts. Furthermore, implementing access control, reliable data sources, data validation, and multi-sig systems help ensure that the information is accurate and trustworthy. Also, we enhance the architecture with robust encryption methods to encrypt the network traffic data before storing it on the blockchain. This will ensure the data is secure and confidential, even if unauthorized parties access the blockchain. In addition, we implement a multi-signature system to modify the smart contract to include a mechanism for requiring multiple signatures before changes can be made to its behavior or data. This is achieved by adding a new function to the smart contract that allows authorized administrators to add or remove other administrators and approve changes to the agreement. The smart contract code [26–28] is presented in the Appendix A.

This code uses the OpenZeppelin library to implement the multi-signature system and the differential privacy mechanism. The Ownable contract provides the only owner modifier, which allows only the contract owner to perform specific actions. The SafeMath library is used for secure arithmetic operations, and the ECDSA library is used for encrypting the network traffic logs using the Elliptic Curve Digital Signature Algorithm.

The NetworkTrafficAnalysis contract has a mapping called logs to store network traffic logs, where the log ID is used as the key, and the encrypted noisy log value is used

as the value. An array called administrators is used to store the addresses of authorized administrators, and a variable called threshold is used to store the multi-signature threshold. Another variable called epsilon is used to store the differential privacy noise.

The administrator and remove administrator functions allow the contract owner to add and remove administrators, respectively. The update threshold and updateEpsilon functions will enable the contract owner to update the multi-signature threshold and the differential privacy noise, respectively.

The store log function is used to store network traffic logs. The log value is first perturbed using differential privacy by adding random noise. The noisy log value is then encrypted using ECDSA, and the number of authorized administrators signed off on the log is checked to ensure that it is greater than or equal to the threshold. Finally, the encrypted noisy log value is stored in the logs mapping, and an event is emitted to indicate that a new log has been held.

The system's architecture is designed to ensure that network traffic logs are securely and transparently stored on the blockchain. The smart contract acts as a decentralized database, where network traffic logs are encrypted using the ECDSA and perturbed using differential privacy to ensure the confidentiality and security of the records.

Overall, the architecture of the system is designed to provide secure, transparent, and tamper-proof storage of network traffic logs, while also ensuring the privacy and security of sensitive information. The system provides a comprehensive solution for network traffic analysis, with features for data storage, encryption, decryption, signature verification, alerts and notifications, and reporting.

5. Use Case

To highlight the real benefits of the proposed BANTA, a use case with real-world data from an enterprise network used, where security and trust are crucial. A centralized network traffic analyzer monitors the network, but there is a risk that malicious actors could compromise the system and tamper with the data. With the BANTA application, each node in the network has a copy of the blockchain, allowing for a decentralized and tamper-proof analysis of the network traffic. Also, the decentralized architecture is highly secure, with no single point of failure. It is tamper-proof and transparent, with data stored across multiple nodes and secured by cryptographic algorithms. In addition, it is scalable and efficient, with data processing distributed across various nodes.

The BANTA quickly notifies the blockchain nodes whenever an intrusion is detected and triggers an alert. This decentralized and secure approach provides a more reliable and trustworthy system for detecting network intrusions than a centralized network traffic analyzer vulnerable to compromise. Blockchain technology also ensures that the data is transparent and auditable, providing a complete and accurate depiction of network activity.

The use case used in this research study is based on the need for security incident management that requires specialized analysis services to comprehend the distributed industry network environment and its possible dangers or vulnerabilities. The vital point in this direction is the classification of network traffic to detect attacks or the forensics of cybercrime. The architecture of the proposed scenario is depicted in Figure 3, simulating an essential distributed industry ecosystem [6,29].

Each industry domain has a distributed node of BANTA, and all of these nodes communicate by the blockchain channel with the central BANTA node.

The scenario aims to identify Tor (The onion router) communications. Tor is a standard method of conveying the most recent, advanced generations of malware and is associated with the Dark Web. Early detection of Tor traffic is essential to help identify malware activities before the final attack.

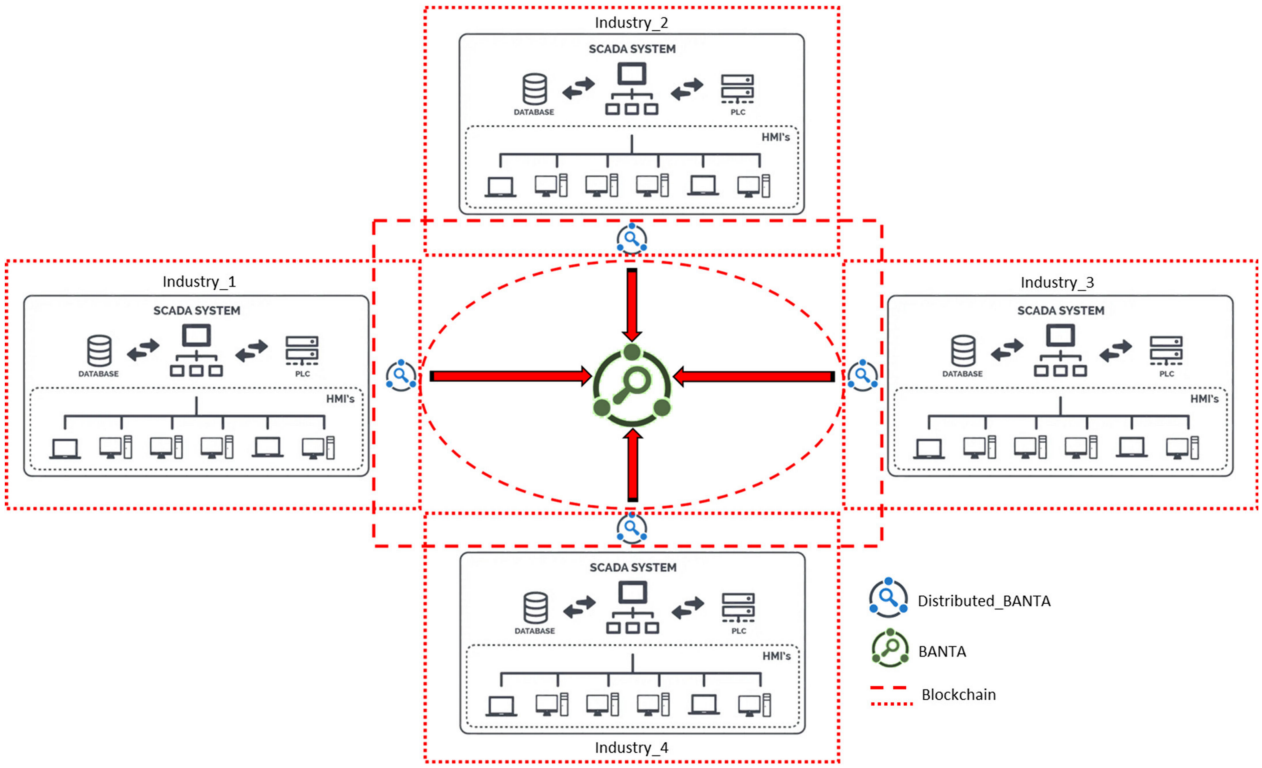

Figure 3. Use case industrial BANTA architecture.

The comparison between ANTA and BANTA is based on the following factors that are considered in a comprehensive evaluation of this use case [30]:

1. Accuracy: The ability of the algorithms to accurately detect patterns and anomalies in network traffic data.
2. Speed: The processing time required to analyze network traffic data using the algorithms.
3. Scalability: The ability of the algorithms to handle large amounts of data and provide results promptly as the volume of network traffic data grows.
4. Flexibility: The ability to modify or update the algorithms as new network traffic patterns and anomalies are discovered.
5. Complexity: The difficulty in implementing, training, and maintaining the algorithms, both from a technical and computational perspective.
6. Resource utilization: The number of computing resources (such as CPU, memory, and storage) required to run the algorithms, as well as any additional hardware or software required.

The ideal solution balances the algorithms' accuracy, speed, and scalability with the complexity, resource utilization, and flexibility required to meet the evaluation criteria. It's essential to consider the specific requirements and priorities of the organization, as well as any constraints or limitations in terms of resources and technology, which is a main important aim of the CTI2SA architecture.

The dataset used for the proposed use case in this study was CICDarknet2020, which includes darknet traffic and matching ordinary traffic from Audio-Stream, Browsing, Chat, Email, P2P, Transfer, Video-Stream, VOIP, Files, Session, and Authentication. Details regarding the dataset and evaluation are available in the relevant literature [6].

The suggested method is a multi-classification issue that attempts to identify and categorize encrypted Tor or VPN communication. The approach for network traffic analysis and feature extraction was founded on the functionality of fundamental network protocols and the acknowledgment technique for safe data transmission and reception. The dataset contains 141,534 data (feature vector) samples, with 93,357 samples classed as non-Tor, 1393 samples classified as Tor, 22,920 samples classified as VPN, and 23,864 samples

classified as other. Each piece contains a flow-id, a class, and 80 features, which include the Source IP Address, Source Port, Destination IP Address, Destination Port, Internet Protocol Version, Timestamp, Duration, and the Total number of packets from source to destination. Cross-validation is used to determine the generalizability of the outcomes of a statistical investigation of a different data set.

Network traffic categorization and analysis is a resampling technique that assesses and trains a model utilizing several iterations and various data components. It is most frequently employed when the purpose is prediction, and the user wishes to determine the practicability of a predictive model. Cross-validation is used to evaluate a model's capacity to predict data that was not included in its estimation, to identify errors such as overfitting and selection bias, and to provide insight into how the model will generalize to an independent dataset. It provides immediate benefits and is often easier to deploy and configure than other systems, making it the ideal starting point for a more proactive security posture. Classifying network traffic enables the organization of network traffic (packets) into traffic classes or groups based on whether or not the traffic matches defined standards. Several network security or service can be enabled by classifying network traffic.

There are three fundamental classification methods for network traffic: port-based, payload-based, and ML-based [7]. Port numbers can identify services capable of handling specific network traffic, whereas payload-based approaches examine the packet's payload for harmful content signatures. Using ML methods to classify data reduces computation costs and enables the quick detection of encrypted communication. Using a corpus of properly annotated examples, ML-based techniques can circumvent the limitations of port- and payload-based systems.

AutoML [31] solutions for real-time monitoring, analysis, and categorization of network traffic represent a significant advancement in securing information systems. AutoML provides methods and processes to enhance ML efficiency and accelerate research by facilitating the establishment of simple, unified interfaces to multiple ML algorithms. Hyperparameter optimization and algorithm configuration are methods for automating the error-prone, time-consuming process of tuning hyperparameters for new tasks. The BANTA architecture considerably enhances CTI2SA's active security features based on the big-data Lambda architecture [32,33]. Lambda is a data-processing architecture that manages massive volumes of data using both batch-processing and stream-processing techniques.

It balances latency, throughput, and fault tolerance by utilizing batch processing to provide complete and accurate views of batch data and real-time stream processing to provide pictures of online data. The real-time or speed layer employs stream processing methods to rapidly index recent data currently unavailable for querying in the batch/serving layers, decreasing the time window of unanalyzable data.

A depiction of the Lambda architecture is presented in the following Figure 4.

The Lambda Architecture is a suggested technique that prevents data inconsistency from frequently occurring in distributed systems. It is built on scale-out, distributed technologies that may be expanded by adding nodes. It is meant to accept and process timestamped events related to existing circumstances instead of overwriting them and to accommodate increasing immutable data collections. The architecture is optimal for large-scale cybersecurity applications.

This system aims to facilitate the adoption of the most accurate ML model that can evaluate network data and respond to vulnerabilities designed to deceive the system. The Lambda BANTA is an autoML solution that efficiently combines a batch engine with historical data to train the ML model. It is a mechanism for automating some of the more complex or mundane operations in the machine-learning lifecycle, including passing data to training algorithms and determining the ideal neural network design for a specific task. In addition, it contributes to the democratization of machine learning by making machine-learning techniques and technology accessible to non-expert users. Hyperparameter optimization is the process of selecting a learning algorithm's optimal hyperparameters.

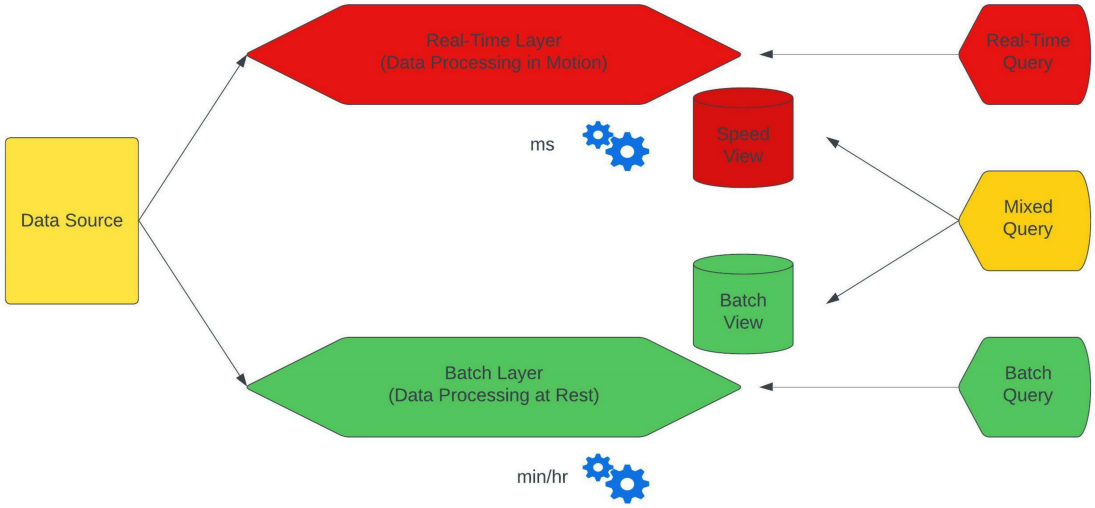

Figure 4. Lambda Architecture.

In the first phase of the proposed model's operation, it is anticipated that the required features will be extracted from the network traffic of each data stream; these data will be saved in the historical data storage and utilized to train the ML model. The data was collected using a hybrid automated IP flow analysis technique whose central modeling concept is based on the open-source framework Stream4Flow [34,35]. Stream4Flow is an open-source platform that offers a complete IP flow analysis solution. It can connect to most IP flow network detectors and has data collection, processing, manipulation, storage, and presentation capabilities. It is scalable and suitable for handling network traffic in various heterogeneous networks. The approach provides IP traffic analysis results with a few-second latency, allowing for real-time investigation of questionable situations.

IPFIXCol (https://github.com/CESNET/ipfixcol2, accessed on 13 January 2023) that the architecture depicted in the Figure 4, is a system for complexly processing IP streams from several sources. It is a flexible flow collector that supports all prevalent network protocols and allows the translation of incoming IP stream data to JSON format. To evaluate bidirectional network traffic flow and remove statistical characteristics and flow data, the open-source CICFlowMeter framework was incorporated as a plugin to the intermediate API. The network traffic analyzer was added to the framework above to extract the main features that determine the nature of network traffic data. New features may be added on an individual basis.

Figure 5 depicts the IPFIXcol architecture, and Figure 6 the proposed architecture con-figured by adding the CICFlowMeter framework as an extension to the original Stream4Flow architecture's intermediate API.

In particular, the framework output consists of six labeled columns for each flow (FlowID, SourceIP, DestinationIP, SourcePort, DestinationPort, and Protocol), from which over eighty network traffic analysis characteristics can be derived. The CICFlowMeter case-specific technique can arbitrarily specify the flow timeout value (e.g., 600 s for both TCP and UDP).

The training process is handled only with autoML, and the winning algorithm with the necessary hyperparameters is sent to the real-time engine for network traffic control. The educational procedure is repeated periodically when the historical data storage grows by 30%. Cross-validation processing and all possible ML approaches are used to retrain the data. The winning algorithm is provided to the real-time engine for hot path control.

ANTA and BANTA use ten popular ML algorithms in the use case that examines. Each architecture's results with the highest categorization accuracy are displayed in the Tables 1 and 2 below.

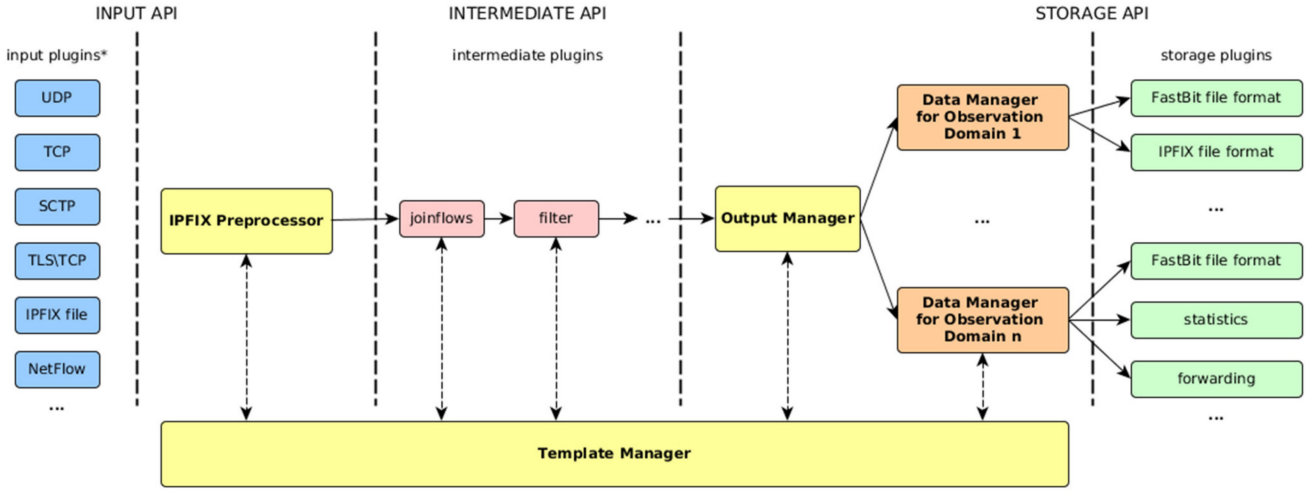

Figure 5. The IPFIXcol Architecture (https://github.com/CSIRT-MU/Stream4Flow, accessed on 13 January 2023). * Select one for IPFIX Preprocessor.

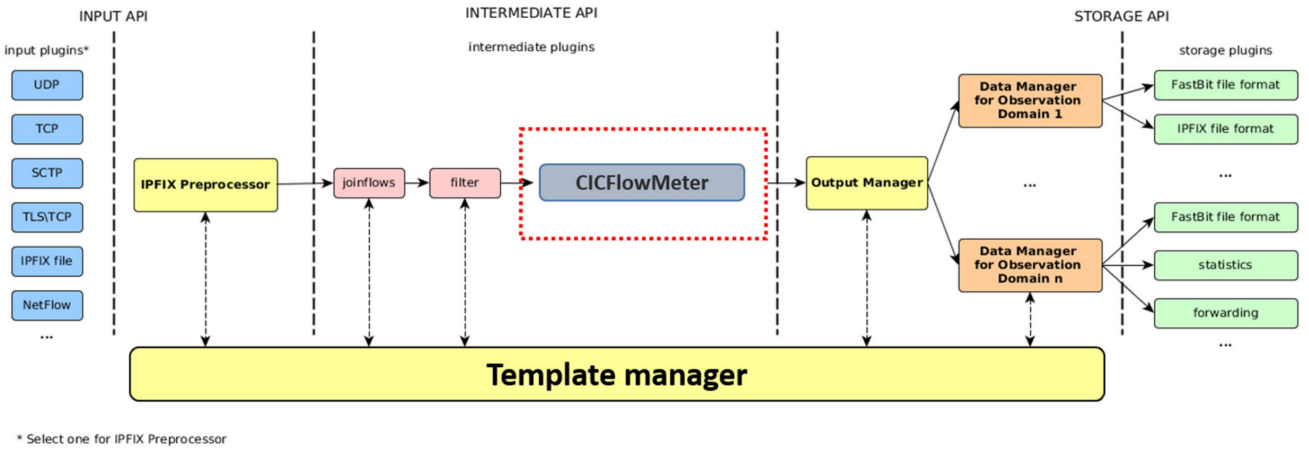

Figure 6. IPFIXcol with CICFlowMeter plugin.

Table 1. Performance metrics of the ANTA method (model selection process).

Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (s)
LGBM	0.9896	0.9997	0.9670	0.9896	0.9896	0.9796	0.9796	10.321
RFC	0.9880	0.9994	0.9581	0.9880	0.9880	0.9765	0.9765	22.577
ETC	0.9875	0.9992	0.9628	0.9875	0.9875	0.9755	0.9755	16.963
DTC	0.9842	0.9902	0.9570	0.9842	0.9842	0.9690	0.9690	3.620
GBC	0.9767	0.9989	0.9397	0.9768	0.9766	0.9542	0.9542	317.461
k-NC	0.9446	0.9823	0.8133	0.9433	0.9435	0.8899	0.8901	19.832
LDA	0.9231	0.9854	0.8510	0.9238	0.9233	0.8494	0.8495	4.570
RC	0.9203	0.8902	0.8307	0.9192	0.9195	0.8424	0.8426	0.280
ABC	0.8525	0.9675	0.7639	0.8770	0.8581	0.7204	0.7262	16.659
QDA	0.8258	0.9662	0.7558	0.8523	0.8151	0.6703	0.6871	2.617

Light Gradient Boosting Machine (LGBM), Random Forest Classifier (RFC), Extra Trees Classifier (ETC), Decision Tree Classifier (DTC), Gradient Boosting Classifier (GBC), k-Neighbors Classifier (k-NC), Linear Discriminant Analysis (LDA), Ridge Classifier (RC), and Ada Boost Classifier (ABC) (QDA).

Table 2. Performance metrics of the BANTA method (model selection process).

Model	Accuracy	AUC	Recall	Prec.	F1	Kappa	MCC	TT (s)
LGBM	0.9912	0.9998	0.9730	0.9910	0.9915	0.9805	0.9804	21.985
GBC	0.9908	0.9996	0.9719	0.9902	0.9880	0.9791	0.9789	34.176
RFC	0.9884	0.9991	0.9702	0.9890	0.9872	0.9790	0.9769	20.442
DTC	0.9860	0.9928	0.9686	0.9811	0.9828	0.9748	0.9699	13.023
ETC	0.9765	0.9917	0.9556	0.9744	0.9700	0.9619	0.9623	421.774
k-NC	0.9400	0.9783	0.9327	0.9562	0.9512	0.8942	0.9009	53.231
ABC	0.9019	0.9133	0.8906	0.9084	0.9093	0.8612	0.8501	14.983
RC	0.8849	0.8890	0.8651	0.8724	0.9002	0.8248	0.8311	20.026
LDA	0.8274	0.8759	0.8391	0.8562	0.8376	0.7084	0.7184	19.942
QDA	0.8083	0.9332	0.7235	0.8397	0.8003	0.6932	0.6784	22.410

Light Gradient Boosting Machine (LGBM), Random Forest Classifier (RFC), Extra Trees Classifier (ETC), Decision Tree Classifier (DTC), Gradient Boosting Classifier (GBC), k-Neighbors Classifier (k-NC), Linear Discriminant Analysis (LDA), Ridge Classifier (RC), and Ada Boost Classifier (ABC) (QDA).

The tables above illustrate the quality and precision of the compared models using a series of assessment measures that represent how well each model performed on the test dataset. It must be underlined that the suggested method employs just the cross-validation data-split method in all training scenarios. The following metrics are displayed in the above table: Accuracy, Area Under the Curve (AUC), Recall, Precision, F1-score, Cohen's kappa coefficient, Matthews Correlation Coefficient (MCC), and Training Time in seconds (TT).

The choice between ANTA and BANTA depends, as mentioned before, on the specific requirements and priorities of the organization, including scalability, flexibility, complexity, and resource utilization. Specifically:

(1) Scalability

Scalability was measured by including the following:

1. Throughput: The amount of data processed by the system over a specific period. This is expressed in data volume (e.g., gigabytes per second), several transactions processed, or several requests served.
2. Latency: The time it takes for a request to be processed and a response to be returned. Lower latency indicates better scalability.
3. Resource utilization: The number of computing resources (such as CPU, memory, and storage) required to process a specific amount of data. Lower resource utilization indicates better scalability.
4. User satisfaction: This is measured by monitoring the time taken to complete a classification task.
5. Error rate: The number of errors or failures during data processing. Lower error rates indicate better scalability.

Table 3 illustrates the scalability of the compared models using the above series of assessment measures that represent how well each model performed on the use case.

Table 3. Scalability performance metrics of the ANTA and BANTA methods.

Model	Throughput	Latency	Resources	User_Sat	Error_Rate
ANTA	583 rps *	56 ms **	73% CPU 56% memory	112 ms	0.0104%
BANTA	861 rps	52 ms	68% CPU 47% memory	94 ms	0.0088%

* rps = requests per second, ** ms = milliseconds.

In evaluating the scalability of the compared models, the following conclusions were met:

1. ANTA: The scalability of the non-blockchain solution depends on the size and performance of the central server or cluster of servers that store and process the data. This solution is typically faster than the blockchain-based solution but is also more vulnerable to a single point of failure or performance degradation.
2. BANTA: The scalability of the blockchain-based solution depends on the size and performance of the network of nodes that make up the blockchain. With more nodes, the answer can handle more data and process it faster, but this also increases the complexity of the network and the resources required to run it.

Based on the above metrics, BANTA is more scalable than ANTA as it has better throughput, lower latency, lower resource utilization, higher user satisfaction, and a lower error rate.

(2) Flexibility:

Flexibility was measured by including the following:

1. Ease of customization: The ability of the system to be easily configured and adapted to meet changing requirements. This includes adding or removing features, changing algorithms, or modifying data processing pipelines.
2. Interoperability: The ability of the system to work with other systems, data sources, and technologies. This includes integrating with existing IT systems or exchanging data with other systems using standard protocols.
3. Upgradeability: The ability of the system to be upgraded with new features and capabilities without disrupting existing operations. This includes adding new nodes to the network, upgrading the software, or incorporating new algorithms.

In evaluating the flexibility of the compared models, the following conclusions were met:

1. ANTA: The flexibility of the non-blockchain solution is limited by the design of the central server or cluster of servers and the algorithms used to analyze the data. Changes to the algorithms or the network architecture can be made more quickly, as there is no need for consensus among multiple nodes.
2. BANTA: The flexibility of the blockchain-based solution is limited by the design of the blockchain, and the algorithms used to analyze the data. Changes to the algorithms or the network architecture can be complex and time-consuming, requiring consensus among the nodes in the network.

(3) Complexity:

Complexity was measured by including the following:

1. The number of components: The number of separate components or elements that make up the system. A system with fewer components is typically less complex.
2. Interconnections: The number of connections and interactions between the components. A system with fewer interconnections is typically less complex.
3. Degree of integration: The extent to which components are integrated and interact. A system with a high degree of integration is typically less complex.
4. Learning Curve: The amount of time and effort required to learn how to use the system effectively. A system with a low learning curve is typically less complex.
5. Error rate: The number of errors or failures that occur during use. A system with a lower error rate is typically less complex.

Table 4 illustrates the complexity of the compared models using the above series of assessment measures that represent how well each model performed on the use case.

Table 4. Complexity metrics of the ANTA and BANTA methods.

Model	Components	Interconnections	Integration	Learning_Curve	Error_Rate
ANTA	equal	equal	equal	0.0003	0.0104%
BANTA				0.0002	0.0088%

In evaluating the complexity of the compared models, the following conclusions were met:

1. ANTA: The complexity of the non-blockchain solution is lower than the blockchain-based solution, as there is only a single central server or cluster of servers to manage and no need for cryptographic security. Training and maintaining the machine learning algorithms is also more straightforward, as it is done on a single centralized entity.
2. BANTA: The complexity of the blockchain-based solution is high due to the need to manage a decentralized network of nodes and secure the data using cryptographic algorithms. Training and maintaining the machine learning algorithms is also complex, as it must be done across multiple nodes in the network.

(4) Resource utilization:

Resource utilization was measured by including the following:

1. CPU utilization: The amount of processing power used by the system. Higher CPU utilization indicates higher resource usage.
2. Memory utilization: The amount of memory used by the system. Higher memory utilization indicates higher resource usage.
3. Disk utilization: The amount of storage used by the system. Higher disk utilization indicates higher resource usage.
4. Network utilization: The amount of network bandwidth used by the system. Higher network utilization indicates higher resource usage.
5. Power consumption: The amount of energy used by the system. Higher power consumption indicates higher resource usage.

Table 5 illustrates the resource utilization of the compared models using the above series of assessment measures that represent how well each model performed on the use case.

Table 5. Resource utilization metrics of the ANTA and BANTA methods.

Model	CPU	Memory	Disk	Network	Power
ANTA	73%	56%	67%	23 Mbps	0.0231 mw
BANTA	78%	67%	42%	18 Mbps	0.0189 mw

mw = milliwatt.

In evaluating the resource utilization of the compared models, the following conclusions were met:

1. ANTA: The resource utilization of the non-blockchain solution is lower than the blockchain-based solution, as there is only a single central server or cluster of servers that must be equipped with the necessary hardware and software to process and store data and run the machine learning algorithms.
2. BANTA: The resource utilization of the blockchain-based solution is high, as each node in the network must be equipped with the necessary hardware and software to process and store data and run the machine learning algorithms.

In final conclusion, the BANTA solution offers greater security and transparency but also comes with increased complexity and resource utilization. The ANTA solution is more straightforward, and requires fewer resources, but is less secure and transparent.

6. Discussion

The proposed solution BANTA uses multi-signature and differential privacy capabilities (Zero-knowledge proofs) to automate network traffic analysis securely and with privacy protection. A significant improvement is that it stores network traffic logs in a decentralized manner, providing transparency and security using blockchain technology. Smart contracts automate the process of network traffic analysis, and a multi-signature system ensures the system's integrity και authentication.

It must be noted that in the context of the smart contract, as presented in Appendix A, formal methods are used to prove the correctness of its behavior, ensure that it satisfies its intended properties, and detect potential vulnerabilities or security risks. By formally verifying the proposed smart contract, developers can increase their confidence in the correctness and security of the system, ultimately improving its reliability and trustworthiness. Specifically, in the case of the sample Solidity contract (Appendix A) for the Network Traffic Analysis system, the following formal methods were used for verification [36,37]:

1. Formal specification language: Specification language is used to formally specify the behavior of the contract and verify its correctness.
2. Model checking: Model checking is used for verifying the correctness of the system by checking all possible executions against a formal specification and ensuring that it satisfies its intended properties.
3. Theorem proving: Theorem proving involves using formal logic to prove the correctness of the system in order to prove properties of the contract, such as safety and liveness properties.
4. Static analysis: Static analysis involves analyzing the program's source code to detect potential errors or vulnerabilities.
5. Runtime verification: Runtime verification involves monitoring the execution of the system to check if it satisfies its formal specification or to detect violations of properties in the contract during execution.

These formal methods are used individually or in combination to comprehensively verify the contract, ensuring its correctness, security, and privacy in the BANTA architecture.

The main differences between ANTA and BANTA are:

1. Architecture: ANTA and BANTA are data-driven systems based on the Lambda architecture, while BANTA enhances the architecture using secure and privacy-preserving techniques.
2. Security and privacy: While ANTA provides security measures for network traffic analysis, BANTA offers additional security and privacy measures through its use of blockchain technology, multi-signature system, and differential privacy algorithms.
3. Data storage: ANTA stores data in a centralized manner, while BANTA stores network traffic logs in a decentralized manner using blockchain.
4. Flexibility: ANTA is a flexible and self-adapting system that can handle massive amounts of data. BANTA is a more rigid system because it uses blockchain technology and smart contracts.
5. Complexity: BANTA is a more complex system due to its use of blockchain technology and smart contracts, while ANTA is a simpler system that focuses on data-driven analysis.

ANTA is a data-driven module focusing on network traffic analysis. ANTA is more flexible and easier to use, while BANTA is more complex but offers more robust security measures. At the same time, BANTA adds a layer of security and privacy through its use of blockchain technology and smart contracts.

Differential privacy algorithms protect sensitive information in the network flow logs while allowing administrators to analyze network traffic without the risk of leakages. By conducting an in-depth analysis of the intrinsic advantages of the proposed method, based on the experimental results presented in the previous section, we could point out the following:

1. Performance impact: BANTA does not consume significant computing resources, leading to latency and reduced network performance.
2. Privacy concerns: BANTA does not violate users' privacy by examining the content of their communications.
3. Cost: BANTA is not expensive to deploy, configure, and maintain. The high cost of specialized hardware, software, and skilled personnel can be a barrier for smaller organizations.
4. Complexity: BANTA is not complex and does not require specialized knowledge to configure and manage. Setting up and maintaining the system is a common challenge for organizations without the necessary expertise.
5. False positives: BANTA generates minimum false positives, indicating that legitimate traffic is malicious. This does not lead to false alarms and unnecessary investigations, wasting time and resources. This is an effect of the system's auto-model-selection method that selects the machine learning model with the most outstanding performance among competitors to maintain the efficacy of the architecture's threat identification capabilities.

In summary, the BANTA solution offers greater security and transparency by leveraging blockchain technology and differential privacy algorithms, which allow for decentralized storage, supports multi-signature systems, and automated network traffic analysis while preserving the privacy of sensitive information. Although the ANTA solution is more straightforward and requires fewer resources, it is less secure and transparent than BANTA. ANTA does not offer the same level of privacy protection as the BANTA solution and it does not benefit from the offerings of the proposed technology incorporated into the specific blockchain proposal. However, ANTA may be suitable for organizations that do not have strict privacy requirements or limited resources.

The choice between cyber defense solutions depends on the organization's specific security, privacy, transparency, and resource utilization requirements. Organizations prioritizing privacy and security may prefer more advanced solutions like the proposed BANTA. In contrast, those with limited resources or less strict privacy requirements may find the common solutions more suitable.

7. Conclusions and Future Research

This study introduces an innovative architectural approach for the intelligent management and mitigation of complex cyber threats. It introduces the BANTA, which improves operational industrial cybersecurity procedures. It is a blockchain-based privacy-preserving system with multi-signature and differential privacy characteristics for automated network traffic analysis.

This innovative study concept has never been presented in the literature before, and we believe it has the potential to enhance the state-of-the-art in ML and blockchain-based industrial cybersecurity dramatically. However, the proposed approach has several limitations that can be improved in future research.

Firstly, the proposed approach assumes that the network administrators and security personnel involved in the multi-signature system are trustworthy. However, in reality, some network administrators may be rogue actors who could compromise the system's security. In future research, this issue could be addressed by implementing additional security measures such as multi-factor authentication or reputation systems to ensure that only trustworthy parties can access the network traffic data.

Secondly, the system's performance may be impacted by the size of the network traffic logs. As the size of the logs grows, the computational complexity of the differential privacy algorithms may increase, leading to slower processing times. In future research, optimization techniques such as indexing or hashing could be implemented to improve the system's performance.

Thirdly, the proposed approach assumes that network administrators have the technical expertise to use smart contracts and the blockchain platform. In future research,

user-friendly interfaces could be developed to make it easier for network administrators with limited technical expertise to use the system.

Lastly, the proposed approach has not been thoroughly tested in real-world scenarios, and unforeseen security and privacy issues may need to be addressed. In future research, large-scale testing and deployment of the system could help identify and address any security and privacy issues that may arise.

The most crucial objective for the evolution of the proposed system is to identify methods for accelerating the convergence of logs and security rules by comparing them. In addition, improving BANTA with more advanced anomaly detection techniques that consider most of the organization's operational factors, such as work schedules, local events, technical upgrades or system adaptations, etc., would significantly improve. It is also beneficial to analyze the system's structure in light of data transformation techniques so that intelligent processes can determine the most efficient ways to represent various forms of structured and unstructured data to facilitate the execution of self-healing rules. The future expansion of BANTA must highlight the use of interpretation models. These models can explain the decision-making process, including the significance of characteristics and the accumulation of local effects. They can specify individual predictions using approaches such as Shapley values. The purpose of model interpretation is to develop human-comprehensible terms for model mechanisms to investigate adversarial attacks and defenses.

Author Contributions: Conceptualization, A.P., A.A. and C.I.; methodology, A.P. and C.I.; software, A.P., C.I., K.D. and K.R.; validation, A.P., A.A., C.I., K.D. and K.R.; formal analysis, A.P., A.A., C.I. and K.R.; investigation, A.P., A.A. and C.I.; resources, A.P. and C.I.; data curation, A.P., A.A., C.I., K.D. and K.R.; writing—original draft preparation, A.P., K.D. and K.R.; writing—review and editing, A.P., A.A., C.I., K.D. and K.R.; visualization, A.P., A.A. and C.I.; supervision, C.I. and K.R.; project administration, A.P.; funding acquisition, A.P., A.A. and C.I. All authors have read and agreed to the published version of the manuscript.

Funding: Co-financed by the European Regional Development Fund of the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH—CREATE—INNOVATE (project code: T2EDK-01469).

Data Availability Statement: The data used in this study are available from the author upon request.

Conflicts of Interest: The authors declare that they have no conflict of interest.

Abbreviations

AutoML	Automated Machine Learning
ANTA	AutoML Network Traffic Analyzer
CTI2SA	Cyber Threat Intelligent Information Sharing Architecture
ML	Machine Learning
BANTA	Blockchained AutoML Network Traffic Analyzer
C&C	Command and Control
IDS	Intrusion Detection Systems
IPS	Intrusion Prevention Systems
BPR	Blind Proxy Redirection
SSL	Secure Sockets Layer
DNS	Domain Name System
IOCs	Indicators of Compromise
WQL	Weaver Query Language
WAN	Weaver Active Nodes
STN	Smart Trade Networks
PoA	Proof of Authority
CTI	Cyber Threat Information
BFT	Byzantine Fault Tolerant
TLS	Transport Layer Security

HSM	Hardware Security Module
ECDSA	Elliptic Curve Digital Signature Algorithm
Tor	The onion router
IPFIXCol	IP Flexible flow Collector
CICFlowMeter	Cluster Ion Counter Flow Meter
FlowID	Flow ID,
SourceIP	Source IP
DestinationIP	Destination IP
SourcePort	Source Port
DestinationPort	Destination Port
LGBM	Light Gradient Boosting Machine
RFC	Random Forest Classifier
ETC	Extra Trees Classifier
DTC	Decision Tree Classifier
GBC	Gradient Boosting Classifier
k-NC	k-Neighbors Classifier
LDA	Linear Discriminant Analysis
RC	Ridge Classifier
ABC	Ada Boost Classifier
AUC	Area Under the Curve
MCC	Matthews Correlation Coefficient
TT	Training Time
mw	milliwatt

Appendix A

Listing A1. Proposed Smart Contract.

```
    pragma solidity ^0.8.0;
    import ‘‘https://github.com/OpenZeppelin/openzeppelin-solidity/contracts/
cryptography/ECDSA.sol’’;
    import ‘‘https:
//github.com/OpenZeppelin/openzeppelin-solidity/contracts/utils/SafeMath.sol’’;
    import ‘‘https:
//github.com/OpenZeppelin/openzeppelin-solidity/contracts/access/Ownable.sol’’;
    contract NetworkTrafficAnalysis is Ownable {
    using SafeMath for uint256;
    using ECDSA for bytes32;
    // Mapping to store network traffic logs
    mapping(bytes32 => uint256) public logs;
    // Array to store authorized administrators
    address[] public administrators;
    // Multi-Signature threshold
    uint256 public threshold;
    // Differential privacy noise
    uint256 public epsilon;
    // Event to emit network traffic logs
    event Log(bytes32 logId, uint256 logValue);
    // Function to add new administrators
    function addAdministrator(address newAdministrator) public onlyOwner {
    administrators.push(newAdministrator);
    }
    // Function to remove administrators
    function removeAdministrator(address oldAdministrator) public onlyOwner {
    uint256 index = administrators.indexOf(oldAdministrator);
    administrators.splice(index, 1);
    }
```

Listing A1. *Cont.*

```
    // Function to update the multi-signature threshold
    function updateThreshold(uint256 newThreshold) public onlyOwner {
    threshold = newThreshold;
    }
    // Function to update the differential privacy noise
    function updateEpsilon(uint256 newEpsilon) public onlyOwner {
    epsilon = newEpsilon;
    }
    // Function to store network traffic logs
    function storeLog(bytes32 logId, uint256 logValue) public {
    // Add differential privacy noise to the log value
    uint256 noisyLogValue =
logValue.add(uint256(keccak256(abi.encodePacked(logId, logValue))) % epsilon);
    // Encrypt the noisy log value using ECDSA
    bytes32 encryptedNoisyLogValue =
bytes32(keccak256(abi.encodePacked(noisyLogValue)));
    // Check if the number of authorized administrators that have signed off on
the log is greater than or equal to the threshold
    uint256 numSignatures = 0;
    for (uint256 i = 0; i < administrators.length; i++) {
    if (ECDSA.recover(keccak256(abi.encodePacked(logId, encryptedNoisyLogValue)),
administrators[i])) {
    numSignatures = numSignatures.add(1);
    }
    }
    require(numSignatures >= threshold, "Not enough authorized administrators
have signed off on this log");
    // Store the encrypted noisy log value
    logs[logId] = encryptedNoisyLogValue;
    // Emit the log event
    emit Log(logId, encryptedNoisyLogValue);
    }
    }
```

References

1. Manogaran, G.; Thota, C.; Lopez, D.; Sundarasekar, R. *Big Data Security Intelligence for Healthcare Industry 4.0*; Springer: Cham, Switzerland, 2017.
2. Mohammed, A.; George, G. Vulnerabilities and Strategies of Cybersecurity in Smart Grid-Evaluation and Review. In Proceedings of the 2022 3rd International Conference on Smart Grid and Renewable Energy (SGRE), Doha, Qatar, 20–22 March 2022; pp. 1–6. [CrossRef]
3. Safavi, S.; Meer, A.M.; Melanie, E.K.J.; Shukur, Z. Cyber Vulnerabilities on Smart Healthcare, Review and Solutions. In Proceedings of the 2018 Cyber Resilience Conference (CRC), Putrajaya, Malaysia, 13–15 November 2018; pp. 1–5. [CrossRef]
4. Nikoloudakis, Y.; Pallis, E.; Mastorakis, G.; Mavromoustakis, C.X.; Skianis, C.; Markakis, E.K. Vulnerability assessment as a service for fog-centric ICT ecosystems: A healthcare use case. *Peer-Peer Netw. Appl.* **2019**, *12*, 1216–1224. [CrossRef]
5. Addeen, H.H.; Xiao, Y.; Li, J.; Guizani, M. A Survey of Cyber-Physical Attacks and Detection Methods in Smart Water Distribution Systems. *IEEE Access* **2021**, *9*, 99905–99921. [CrossRef]
6. Drias, Z.; Serhrouchni, A.; Vogel, O. Analysis of cyber security for industrial control systems. In Proceedings of the 2015 International Conference on Cyber Security of Smart Cities, Industrial Control System and Communications (SSIC), Shanghai, China, 5–7 August 2015; pp. 1–8. [CrossRef]
7. Goli, Y.D.; Ambika, R. Network Traffic Classification Techniques—A Review. In Proceedings of the 2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS), Belgaum, India, 21–22 December 2018; pp. 219–222. [CrossRef]
8. Lim, K.-S.; Stadler, R. Real-time views of network traffic using decentralized management. In Proceedings of the 2005 9th IFIP/IEEE International Symposium on Integrated Network Management, 2005. IM 2005, Nice, France, 19 May 2005; pp. 119–132. [CrossRef]
9. Ageyev, D.; Radivilova, T.; Mulesa, O.; Bondarenko, O.; Mohammed, O. Traffic Monitoring and Abnormality Detection Methods for Decentralized Distributed Networks. In *Information Security Technologies in the Decentralized Distributed Networks*; Oliynykov, R., Kuznetsov, O., Lemeshko, O., Radivilova, T., Eds.; Springer International Publishing: Cham, Switzerland, 2022; pp. 287–305. [CrossRef]

10. Dongre, V.C.; Shikalpure, S.G. Ensuring privacy preservation in wireless networks against traffic analysis by employing network coding and Blowfish encryption. In Proceedings of the 2016 International Conference on Signal and Information Processing (IConSIP), Nanded, India, 6–8 October 2016; pp. 1–5. [CrossRef]
11. Yao, Y.; Chang, X.; Li, L.; Liu, J.; Wang, H. Dual Privacy-Preserving Lightweight Navigation System for Vehicular Networks. *IEEE Access* **2022**, *10*, 121120–121132. [CrossRef]
12. Wilbur, M.; Dubey, A.; Leão, B.; Bhattacharjee, S. A Decentralized Approach for Real Time Anomaly Detection in Transportation Networks. In Proceedings of the 2019 IEEE International Conference on Smart Computing (SMARTCOMP), Washington, DC, USA, 12–15 June 2019; pp. 274–282. [CrossRef]
13. Zhang, P.; Sun, S. Decentralized Network Anomaly Detection via a Riemannian Cluster Approach. In Proceedings of the GLOBECOM 2017-2017 IEEE Global Communications Conference, Singapore, 4–8 December 2017; pp. 1–6. [CrossRef]
14. Guo, J.; Ding, X.; Wu, W. Reliable Traffic Monitoring Mechanisms Based on Blockchain in Vehicular Networks. *IEEE Trans. Reliab.* **2022**, *71*, 1219–1229. [CrossRef]
15. Cao, S.; Foth, M.; Powell, W.; Miller, T.; Li, M. A blockchain-based multisignature approach for supply chain governance: A use case from the Australian beef industry. *Blockchain Res. Appl.* **2022**, *3*, 100091. [CrossRef]
16. Chen, Y.; Dai, H.; Yu, X.; Hu, W.; Xie, Z.; Tan, C. Improving Ponzi Scheme Contract Detection Using Multi-Channel TextCNN and Transformer. *Sensors* **2021**, *21*, 6417. [CrossRef]
17. Papanikolaou, A.; Alevizopoulos, A.; Ilioudis, C.; Demertzis, K.; Rantos, K. A Cyber Threat Intelligence Management Platform for Industrial Environments. *arXiv* **2023**, arXiv:2301.03445. [CrossRef]
18. Chatziamanetoglou, D.; Rantos, K. CTI Blockchain-Based Sharing using Proof-of-Quality Consensus Algorithm. In Proceedings of the 2021 IEEE International Conference on Cyber Security and Resilience (CSR), Rhodes, Greece, 26–28 July 2021; pp. 331–336. [CrossRef]
19. Alsaffar, N.; Ali, H.; Elmedany, W. Smart Transportation System: A Review of Security and Privacy Issues. In Proceedings of the 2018 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), Sakhier, Bahrain, 18–20 November 2018; pp. 1–4. [CrossRef]
20. Boussada, R.; Elhdhili, M.E.; Saidane, L.A. A survey on privacy: Terminology, mechanisms and attacks. In Proceedings of the 2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA), Agadir, Morocco, 29 November 2016–2 December 2016; pp. 1–7. [CrossRef]
21. Coulter, R.; Han, Q.-L.; Pan, L.; Zhang, J.; Xiang, Y. Data-Driven Cyber Security in Perspective—Intelligent Traffic Analysis. *IEEE Trans. Cybern.* **2020**, *50*, 3081–3093. [CrossRef] [PubMed]
22. Rantos, K.; Drosatos, G.; Demertzis, K.; Ilioudis, C.; Papanikolaou, A. Blockchain-Based Consents Management for Personal Data Processing in the IoT Ecosystem. 2022. Available online: https://www.scitepress.org/PublicationsDetail.aspx?ID=+u1w9%2fItJqY%3d&t=1 (accessed on 17 April 2022).
23. Aich, S.; Chakraborty, S.; Sain, M.; Lee, H.; Kim, H.-C. A Review on Benefits of IoT Integrated Blockchain based Supply Chain Management Implementations across Different Sectors with Case Study. In Proceedings of the 2019 21st International Conference on Advanced Communication Technology (ICACT), PyeongChang, Republic of Korea, 17–20 February 2019; pp. 138–141. [CrossRef]
24. Bai, L.; Liu, L. Research on Software Defined Network Security Model Based on Blockchain. In Proceedings of the 2021 6th International Conference on Intelligent Computing and Signal Processing (ICSP), Xi'an, China, 9–11 April 2021; pp. 150–153. [CrossRef]
25. Aleksieva, V.; Valchanov, H.; Huliyan, A. Implementation of Smart-Contract, Based on Hyperledger Fabric Blockchain. In Proceedings of the 2020 21st International Symposium on Electrical Apparatus Technologies (SIELA), Bourgas, Bulgaria, 3–6 June 2020; pp. 1–4. [CrossRef]
26. Bartolucci, S.; Fiorentino, S. Blockchain and Smart Contracts as New Governance Tools for the Sharing Economy. In Proceedings of the 2021 IEEE 18th International Conference on Software Architecture Companion (ICSA-C), Stuttgart, Germany, 22–26 March 2021; pp. 118–119. [CrossRef]
27. Demertzis, K.; Iliadis, L.; Tziritas, N.; Kikiras, P. Anomaly detection via blockchained deep learning smart contracts in industry 4.0. *Neural Comput. Appl.* **2020**, *32*, 17361–17378. [CrossRef]
28. Wang, S.; Ouyang, L.; Yuan, Y.; Ni, X.; Han, X.; Wang, F.-Y. Blockchain-Enabled Smart Contracts: Architecture, Applications, and Future Trends. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *49*, 2266–2277. [CrossRef]
29. Conti, M.; Donadel, D.; Turrin, F. A Survey on Industrial Control System Testbeds and Datasets for Security Research. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 2248–2294. [CrossRef]
30. Al Jallad, K.; Aljnidi, M.; Desouki, M.S. Anomaly detection optimization using big data and deep learning to reduce false-positive. *J. Big Data* **2020**, *7*, 68. [CrossRef]
31. Feurer, M.; Klein, A.; Eggensperger, K.; Springenberg, J.; Blum, M.; Hutter, F. Efficient and Robust Automated Machine Learning. In *Advances in Neural Information Processing Systems*; NeurIPS: New Orleans, LA, USA, 2015; Volume 28. Available online: https://papers.nips.cc/paper/2015/hash/11d0e6287202fced83f79975ec59a3a6-Abstract.html (accessed on 29 May 2022).
32. Alghamdi, R.; Bellaiche, M. A Deep Intrusion Detection System in Lambda Architecture Based on Edge Cloud Computing for IoT. In Proceedings of the 2021 4th International Conference on Artificial Intelligence and Big Data (ICAIBD), Chengdu, China, 28–31 May 2021; pp. 561–566. [CrossRef]

33. Suthakar, U.; Magnoni, L.; Smith, D.R.; Khan, A. Optimised Lambda Architecture for Monitoring Scientific Infrastructure. *IEEE Trans. Parallel Distrib. Syst.* **2021**, *32*, 1395–1408. [CrossRef]
34. Jirsik, T. Stream4Flow: Real-time IP flow host monitoring using Apache Spark. In Proceedings of the NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium, Taipei, Taiwan, 23–27 April 2018; pp. 1–2. [CrossRef]
35. Jirsik, T.; Celeda, P. Toward real-time network-wide cyber situational awareness. In Proceedings of the NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium, Taipei, Taiwan, 23–27 April 2018; pp. 1–7. [CrossRef]
36. Almakhour, M.; Sliman, L.; Samhat, A.E.; Mellouk, A. Verification of smart contracts: A survey. *Pervasive Mob. Comput.* **2020**, *67*, 101227. [CrossRef]
37. Krichen, M.; Lahami, M.; Al, Q.A. Formal Methods for the Verification of Smart Contracts: A Review. In Proceedings of the 2022 15th International Conference on Security of Information and Networks (SIN), Sousse, Tunisia, 11–13 November 2022; pp. 1–8. [CrossRef]