Blockchained Adaptive Federated Auto MetaLearning BigData and DevOps CyberSecurity Architecture in Industry 4.0

Konstantinos Demertzis[1,2(✉)], Lazaros Iliadis[1], Elias Pimenidis[3], Nikolaos Tziritas[4], Maria Koziri[4], and Panagiotis Kikiras[4]

[1] School of Civil Engineering, Democritus University of Thrace, Kimmeria, Xanthi, Greece
kdemertz@fmenr.duth.gr, liliadis@civil.duth.gr
[2] Department of Physics, International Hellenic University, St. Loukas, 65404 Kavala, Greece
[3] Computer Science and Creative Technologies, University of the West of England, Bristol, UK
Elias.Pimenidis@uwe.ac.uk
[4] Department of Computer Science, University of Thessaly, 35100 Lamia, Greece
{nitzirit,mkoziri,kikirasp}@uth.gr

Abstract. Maximizing the production process in modern industry, as proposed by Industry 4.0, requires extensive use of Cyber-Physical Systems (CbPS). Artificial intelligence technologies, through CbPS, allow monitoring of natural processes, making autonomous, decentralized and optimal decisions. Collection of information that optimizes the effectiveness of decisions, implies the need for big data management and analysis. This data is usually coming from heterogeneous sources and it might be non-interoperable. Big data management is further complicated by the need to protect information, to ensure business confidentiality and privacy, according to the recent General Data Protection Regulation - GDPR. This paper introduces an innovative holistic Blockchained Adaptive Federated Auto Meta Learning Big Data and DevOps Cyber Security Architecture in Industry 4.0. The aim is to fill the gap found in the ways of handling and securing industrial data. This architecture, combines the most modern software development technologies under an optimal and efficient framework. It successfully achieves the prediction and assessment of threat-related conditions in an industrial ecosystem, while ensuring privacy and secrecy.

Keywords: Blockchain · MetaLearning · Federated learning · CyberSecurity · Privacy · Industry 4.0 · GDPR

1 Introduction

According to the Industry 4.0 standard [1], cyber-physical systems play an active role in monitoring and supervising natural processes, within partially structured smart factories. In this way, they create a virtual copy of the physical world (digital twins) and they make autonomous decentralized decisions, in order to maximize the production process. Through the Industrial Internet of Things (IIoT) business network, these systems

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021
L. Iliadis et al. (Eds.): EANN 2021, INNS 3, pp. 345–363, 2021.
https://doi.org/10.1007/978-3-030-80568-5_29

communicate and collaborate in real time, to implement all kinds of smart production solutions, organizational services and operational processes which offered and used by participants in the production chain [2]. This vision allows the manufacturing sector to make huge innovative leaps, to gain significant extroversion and to develop activities that were previously impossible. However, the complete transformation of the supply chain to a truly integrated and fully automated process, requires the continuous and endless collection of data, from each stage of the production scale. The target is to determine the current situation and to investigate the history of each individual stage of the chain. The process of collecting and analyzing information is optimized with the continuous data flow and it offers the production process an accurate picture of the effectiveness of decisions and their execution. This implies the continuous production of large-scale data which come from heterogeneous sources [3].

Intelligent Real-Time Data Analysis Systems (IRTAS), have the ability to demonstrate experiential learning and optimal decision-making skills without human intervention. Of course, they should have been properly trained by historical data sets, which are representative of the problem they are trying to solve. IRTAS are considered most suitable for use in industrial environments, which are characterized by strong interdependencies and by a high degree of complexity in terms of vulnerabilities and threats. But even these intelligent algorithms are challenged and controlled by a variety of possible factors, which are mainly related to securing privacy of trade secrets. It is a fact that existing technologies fall short of best practices against emerging risks. Some of the most important problems that arise during knowledge mining are related to the speed at which information arrives and the natural tendency of data to evolve over time. This results in the degradation of classifiers due to the constant change of information (concept drift) [4]. Many of the learning methods used for data analysis, have a "memory erase" property that allows a time frame to be set, based on which learning methods "forget" the past. In this way, the gradual attenuation of the importance of past data is achieved. This feature allows the model to operate safely even in the event of sudden changes, that may indicate trends or even noise in the input data, e.g. outliers or extreme values. However, this feature is also associated with a serious drawback of flow analysis methods called "Catastrophic Interference" or "Catastrophic Forgetting" [5]. This concerns the tendency of an algorithm to completely forget the information it had learned during its training process, in the cases of mnemonic connection models, resulting in its degradation and the gradual devaluation of its predictive power.

A good way to tackle this challenge is to define the problem of federated learning [6] and to design the data pipeline so that labels are collected silently. This can be done through user interactions, by providing feedback on models' responses based on specific actions, or activated events. This challenge parallels the GDPR guidelines [7] and the introduced general bans on safeguarding and protecting privacy.

2 Motivation

The innovation of this paper is the fact that it introduces an innovative Cyber Security architecture for use in Industry 4.0 applications. Prototyping the proposed architecture by combining the most up-to-date methods, can integrate specialized processes towards

the development of security applications for modern information systems. This can be achieved through an adaptable flexible and easy-to-use operating environment.

The proposed solution involves the development of a *Block-chained Adaptive Federated Auto Meta-Learning (BLAFAM-BD^{2}C) Big Data and DevOps Cyber Security Architecture* in Industry 4.0, towards digital security and privacy, which is based on the Industry 4.0 model. This method will classify network traffic coming from IIoT, aiming to identify abnormalities due to cyber-attacks. The characteristics of the proposed system and in particular its architecture, allow the analysis, forecasting, monitoring and management of complex situations related to information systems security. This implements a hybrid system with the most technologically advanced computational intelligence methods. The design of the system is based on a multi-layered, modular architecture and distributed design, which allows its interconnection with existing monitoring systems, as well as its expansion through the integration of new functions. Thus the system can be effective in complex environments of high complexity. Its modular architecture is characterized by the abilities to scale at multiple levels and to be integrated into heterogeneous environments, thus ensuring high interoperability.

The aim of the *BLAFAM-BD^{2}C* architecture is the implementation of a modern and technologically up-to-date distributed information intelligence information system, for the timely and valid study of digital threats [8, 9]. This is implemented with fully automated methods of modern computational intelligence, which can fully respect privacy and protect industrial confidentiality [10, 11]. The goal is intelligent data analysis, forecasting and detection of dangerous situations, providing advanced privacy services in distributed environments. More specifically, it aims in further processing or analysis of information from computer systems, ensuring that this information is reliable, preserving and safely storing, organizing them to ensure easy updating and introducing new data, introducing a mechanism for ensuring industrial secrecy and performing parallel provision of only relevant information, in order to further support decision-making systems.

3 Literature Review

This chapter, makes a brief presentation of the published work on methodologies and architectures that can perform analysis of big data, along with their applications in distributed environments. Moreover, it discusses techniques for detecting anomalies in industrial environments [12, 13] as well as methods of ensuring privacy and industrial secrecy. Network traffic analysis is a key and promising tool, which has been widely used in dealing with digital threat [14]. Its contribution to the application of machine learning methods is considered important [15]. It uses features derived from both static and dynamic analysis [16, 17], to detect malicious applications, to perform categorization of web traffic, to analyze malware traffic and to detect botnets. Several authors suggest new detection techniques, by introducing different methodologies of classifying network activity.

Hsu et al. [18] proposed a real-time system for locating botnets based on delayed HTTP/HTTPS request anomalies. This approach, although accompanied by very promising results, is an unequivocal effort to tackle digital threats, which does not allow generalization to more complex types of attacks or threats. More complex algorithmic standardizations are proposed in [19], which uses ensembles and combinations of AdaBoost, Hidden Markov, Naive Bayesian and Maximum Entropy, to perform detailed classifications of network traffic. They are using few bytes of payload which makes it difficult to reveal the true dimension of digital threats. The work of Alshammari et al. [20] combines computational intelligence techniques to classify web traffic in order to investigate digital threats. Unfortunately, this approach uses minimal technical data that can clearly identify the type and size of a threat, and even worse, payload data, IP addresses and port numbers are not considered. It is a fact that many new applications have been implemented for the optimal management of large-scale data with the application of the Lambda architecture [21]. A very interesting work was presented by Lee et al. [22] which proposes intelligent data management with two state estimation algorithms. The first, uses the ability to monitor events in a finite measurement window, while the latter asymptotically improves data flows, by eliminating noise and disturbances, restoring the system to a proper state. This work implements a system that remains theoretically stable in a state of proper operation determined by experts, but which is not able to automatically adapt to dynamic changes or situations of marginal noise and disturbances.

Chen et al. [23] propose a secure learning system for malware detection, which adopts classifier retraining techniques as well as methods for normalizing security limits. This approach does not adapt itself, requiring retraining, with all the disadvantages that this may entail, such as computer costs, resource consumption, delays, dead time. Combined techniques such as bagging [24] or adaptive windowing have been used to deal with evolving data streams and their optimal handling. A typical example is the work of [25] which introduces methodologies for early shift detection which can be used to detect anomalies. These methods can determine the evolution of a system, but they are not an active security approach, as they fail to in-depth control the content of web traffic and to reveal complex attack techniques. Specializing in the industrial environment and in the respective proposed security systems, Chen and Abdelwahed, [26] have implemented Machine Learning (ML) to monitor the performance of industrial SCADA control systems. This approach proactively assesses impending attacks on a given natural infrastructure system model. It is based on classical ML models, which are trained at a certain time, using a specific set of data. Their predictive ability is based solely on the success of training. This implies a serious inability to generalize. Based on the same reasoning and keeping the respective weaknesses, Soupionis et al. [27] proposed a combined method for the automatic detection and classification of cyber-attacks occurring in the electrical grid system.

Regarding privacy and industrial secrecy, the research presented in [28] introduces an IIoT decentralized peer-to-peer platform, based on blockchain technology, where communication is allowed without the need for a trusted intermediary. However information is not encrypted when used by a third party peer. Smart contracts, which perform predefined and pre-agreed procedures between participants in an industrial blockchain network, also apply in research [29]. This system proves to be very efficient, as the

users can monitor how their electricity is used, providing information on a platform that cannot be manipulated by any party. A serious weakness of the proposed system is the lack of techniques for automatically redefining smart contracts, based on newer needs or services. Finally, Liopis et al. [30] presented a comparative analysis between visualization techniques, using the concept of business image, to support and handle digital events. The authors present the benefits and the general way of using the technologies in question, while they expect that the visualization could be significantly enhanced by the use of computational intelligence algorithms.

4 The Proposed IIoT Framework

Maintaining knowledge of the industrial environment according to the Industry 4.0 standards is a serious and ongoing challenge. As technologies, automation and communication are developing with astonishing speed, information is a critical point for the continuation of its activities. In this context, the IIoT ecosystem is adopted under the Industry 4.0 standard, for the purpose of construction automation, remote mechanical diagnosis and predictive management of industrial automation functionality and supply chain control. The above actions are achieved with remote sensing capabilities, remote extensive data collection-processing and by direct or indirect communication between interconnected devices and applications. To achieve this communication and the overall satisfaction of services that requires high availability connectivity, large bandwidth and extremely low latency, the IoT ecosystem has incorporated a huge range of wired and wireless high speed interconnection technologies. Respectively, it has adopted a set of infrastructures and models of network architecture. This will achieve the flexible and scalable use of cloud applications, data transfer and machine-to-machine communication, across the path of thousands of devices that can integrate. The usual automatic control devices in the IIoT ecosystem are the SCADA systems, which are implemented based on Programmable Logic Controller (PLC) devices and the sensors used in the control loops to collect all kinds of measurements. The above systems, which are active devices of the infrastructure network, are properly interconnected to allow remote monitoring and control of processes with high response speeds. This is achieved even in cases where the devices are distributed between different remote points. This ecosystem with the specialized processes it performs, has introduced a large number of new threats, beyond its significant benefits and upgrades. These threats are mainly related to the specific purposes that they perform, to their different design specifications, to their specialized communication protocols and to the heterogeneous devices that they are called to interconnect.

Threats, and in particular digital attacks on critical infrastructure in Industry 4.0, are referred to as *Advanced Persistent Threats* (APTs) [31], as they can take over mechanical control, dynamic rearrangement of centrifuges or reprogramming of devices. This is done in order to speed up or slow down their operations, leading to the destruction or permanent damage of the entire industrial equipment. The cybercriminals who direct them, are fully acquainted with specialized methods and tools for the exploitation of unknown vulnerabilities (zero days) [32] and they try not to be perceived without taking into account time. Most of the time, they are extremely capable, organized, funded and

have significant incentives. In these cases, any anomaly detection in the operation of the devices in question is extremely important as it can reveal an ongoing attack.

Under this spirit, this paper introduces *an innovative architecture,* which is based on the most advanced technologies and ensures in the most efficient and intelligent way, the secure network communication between the traded devices in the industry 4.0 standard. More specifically, this paper proposes the *BLAFAM-BD*^{2}*C* in Industry 4.0. This approach implements programmable, intelligent control of network traffic, to detect anomalies that have good chances to be related to APT attacks.

It is an intelligent mechanism for monitoring and detecting abnormalities in the communication network between the devices of the Industry 4.0 environment. This is done based on a system performing automatic analysis of the digital packets of the network's traffic. This creates an automated intelligent neural network that can control and locate abnormalities, train and update the model with federal learning and communicate to all involved devices, through a distributed blockchain system. The method's architecture is described as follows:

1. When a device wants to communicate with another, the proposed intelligent approach is activated, which implements a mechanism for the control of network's traffic aiming to detect anomalies.
2. In the first phase, the features of the network's traffic are exported. These are the input vectors to a neural network which is automatically created with the Neural Architecture Search technique.
3. The model is first trained on the host server with some initial data. This is done in order to initiate the training process.
4. Then the model is encrypted with *homomorphic* encryption and via *blockchain* it is sent to the nodes that will use it.
5. The nodes in question take the model and they improve it by using the data at their disposal.
6. The improved model is encrypted and it is returned via *blockchain* to the server.
7. The aggregation of the optimal models is performed and the weighted average is obtained by using the *Grid Search Weighted Average Ensemble* method [33].
8. The resulting final model is returned via blockchain back to the final nodes.

With this architecture, even if the initial data is not appropriate, there is a continuous improvement of the model so that it can categorize with great accuracy the anomalies in the network's traffic. If the traffic is considered normal, further communication is allowed, while otherwise, communication is not allowed and an alarm is sent to the control centre. Further checks are done. The above implementation, enables the creation of a secure distributed platform that can control and complete the associated transactions in the Industry 4.0 standard, even from terminals (edge computing). This continuously improves the capabilities and reliability of control methods. The use of adaptive learning technologies is an intelligent solution to detect unsafe network traffic anomalies related to digital security incidents. At the same time, this logic provides automated solutions and procedures for real-time monitoring of critical infrastructure related to the high availability of Industry 4.0 technology [1].

The design of the system is based on a multi-layered, modular architecture and distributed design, which allows its interconnection with any existing system, and its expansion through the integration of new functions. Such a system is effective in heterogeneous environments. In this way, it is not enough just to record the data, to visualize the situation and to predict its evolution, using a standardized template. Utilizing multiple technologies, it offers tools and possibilities for composing an adaptive system which can act autonomously. Moreover, each one of its subsystems can fully and optimally meet the arising needs and requirements.

This research proposes a hybrid technique for automated IP flow analysis, whose basic idea of standardization and operation is based on the *open source Stream4Flow* architecture [34, 35]. The most advanced data processing technologies, network traffic monitoring and real-time visualization are used. Specifically, *Stream4Flow* offers a complete stack solution for IP flow analysis, where most IP streaming network detectors can be connected. Tools for data collection, processing, handling, storage and presentation can be integrated. Thanks to the scalability of the framework, it is suitable for processing network traffic in a wide range of heterogeneous networks of scalable capabilities, while its distributed nature allows large-scale computationally intensive analysis. The whole idea is based on the *Stream4Flow* framework, which is formed by the IPFIXCol (*IP Flow Information Export Protocol*) [36] collector, the message exchange system Kafka [37], the Apache Spark [38] and the Elastic Stack [39]. A general idea of its architecture is presented in Fig. 1, below.

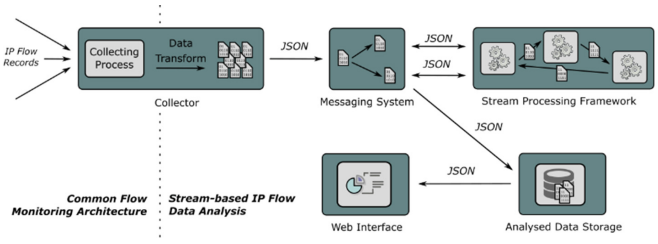

Fig. 1. Stream4Flow architecture (https://stream4flow.ics.muni.cz/)

In general, IPFIXCol allows the conversion of incoming IP stream records into JSON (JavaScript Object Notation) format provided on the *Kafka* messaging system. The *Kafka* adds serious scalability and distribution capabilities, which provide adequate data performance. The *Apache Spark* is used as a feed processing framework for fast IP data flow, providing interoperability in the most popular programming languages (Scala, Java, Python), where it fully supports the programming model *MapReduce*. The results of the analysis are stored in *Elastic Stack* containing *Logstash, Elasticsearch* and *Kibana*, which allow the results to be stored, searched and visualized. The framework also contains an additional web interface to facilitate management and display the complex results of the analysis. Utilizing the capabilities of Stream4Flow, serious additions−improvements in its architectural modelling were studied and developed in order for it to respond optimally

to specialized threats. The information system is described in Fig. 2 below. The individual parts of the proposed architecture and their mode of operation are presented in detail below.

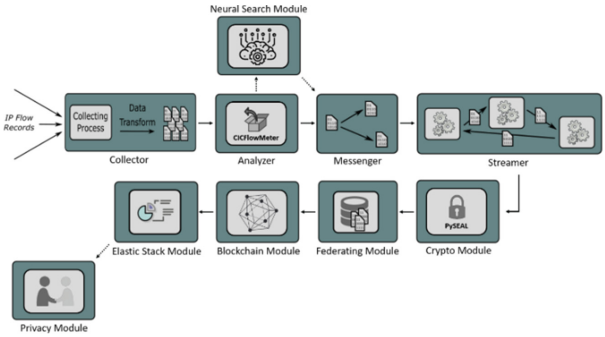

Fig. 2. Proposed architecture

4.1 Collector

The IPFIXcol is a framework for the complex processing of Internet Protocol (IP) flows from multiple different sources. Its purpose is to provide a flexible collector that fully supports the IPFIX protocol including other components. A view of the introduced architecture is illustrated in Fig. 3 below:

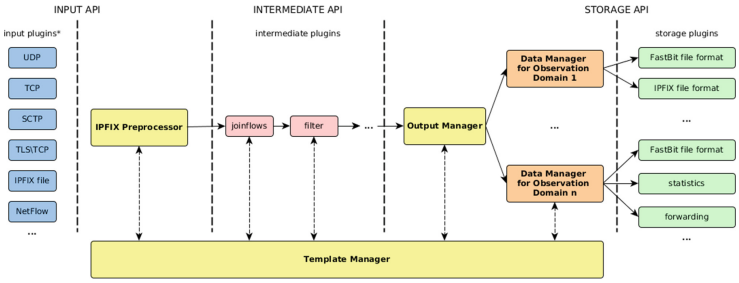

Fig. 3. IPFIXcol architecture (https://stream4flow.ics.muni.cz/)

The collector kernel supports input, intermediate, and output plugins for receiving, processing, and storing streaming data.

4.2 Analyzer

A web traffic analyzer has been added to this box to extract the most important features that can determine the nature of the information contained in web traffic. Moreover, the open source framework CICFM FlowMeter was added as a plugin to the intermediate API. The ICICFM (Intercell Interference Coordination Flow Meter) was used in this

specific application as an analyzer of the network's bi-flow traffic flow. It can export statistical characteristics and flow information. New features can also be added on a case-by-case basis, as well as precisely controlling the duration of a stream. TCP streams are usually terminated during connection when the FIN packet is received, while UDP streams are terminated following a flow time limit. (When either side of a TCP data transmission is done, a FIN signal is sent to close the connection).

In the ICICFM, the flow time limit value can be arbitrarily assigned by the individual schema of the case under consideration e.g. 600 s for both TCP and UDP. In particular, the output of the frame includes six columns, labeled for each flow (FlowID, SourceIP, DestinationIP, SourcePort, DestinationPort and Protocol) while more than 80 traffic analysis features can be exported. A view of the proposed architecture, as formatted by adding the CICFlowMeter framework as an add-on to the intermediate API of the original Stream4Flow architecture is illustrated in Fig. 4:

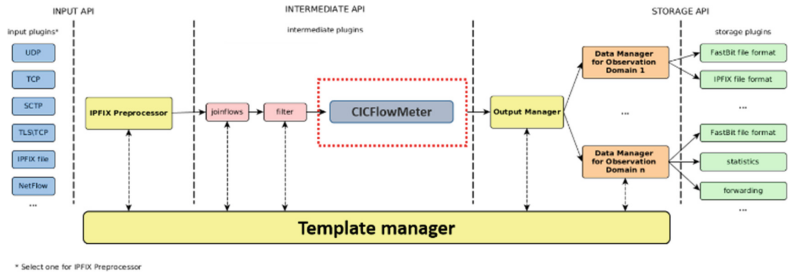

Fig. 4. The proposed IPFIXcol architecture with ICICFlowMeter

4.3 Neural Search Module

The Neural Search Module was added to the proposed architecture, in order to create an automatic post-learning system to be used for the automatic initial development of the learning model, aiming to detect network anomalies. The most appropriate meta-learning architecture for neural networks is Neural Architecture Search (NAS) [40]. It is an automated learning technique for automating the design of artificial neural networks. NAS's main goal is to automate the process of finding the architecture of the best neural network for a data set. The methodology followed to develop architectures of neural networks capable of solving a given problem in NAS, concerns the delimitation of the search space, which includes virtually all possible hyperparameters that can be tuned based on the data used. The design of the NAS strategy [41] was based on Auto-Keras, an open source alternative to Google's AutoML.

4.4 Messenger

Apache Kafka (a distributed software platform designed to process data flows) [37] is used as a messaging system to subscribe and publish them. Based on its architecture,

the storage level is essentially a queue for publishing and consuming messages, with enormous scalability, designed as a distributed transaction log. Incoming streams are organized into topics, where essentially either the information sent or the information to be read, must belong to a specific topic. The data enter the different topics in the form of records, where each record contains a key, (a value which is essentially the information involved) and a time stamp.

Apache Kafka, due to its distributed nature is implemented in clusters called *Kafka brokers*. The topics are divided into partitions in order to have parallelism. It is a fact that multiple consumers are allowed to belong to the same topic. Splitting and sharing data into different brokers, allows different consumers to read the same topic in parallel or to read different partitions in parallel. This creates the conditions for particularly high data rates and processing. Error tolerance is achieved by having replicates of the partitions of each topic in various brokers, where one holds the status of the leader. If for some reason there is a failure on the part of the leader, a copy will take its place and the work will not be interrupted.

4.5 Streamer

Spark Streaming is used to manipulate and analyze data streams [38]. It is an extension of the Spark API kernel that enables scalable, high-performance, error-tolerant data processing. The data can originate from many sources such as Kafka, Kinesis [42] or TCP sockets and they can be processed using complex algorithms, that are expressed with high-level functions, such as map, reduce, join and window. The processed data can be forwarded to file systems, databases and real-time control panels, as shown in the Fig. 7 below. The basic structure that Spark uses to transfer data from disk to memory is Resilient Distributed Datasets (RDDs). They are immutable structures that can be shared in parallel at the nodes of a cluster of computers and contain data in various formats. Respectively *Spark Streaming* takes input data streams and divides them into batch data as presented in the following Fig. 5:

Fig. 5. Proposed apache spark streaming (https://spark.apache.org/)

Its main advantage is that it provides high-level capabilities for combining and integrating different types of calculations, which otherwise would require the use of separate distributed systems. At the same time it automates and hides from users, important low level details. In the proposed architecture, *Streamer* offers a high-level abstract entity of discrete flows, derived from the *Messenger*, which is said to use *Apache Kafka.*

4.6 Crypto Module

To create a robust mechanism for controlling the possible leakage of confidential data, when data is stored, transmitted or calculated, even in cases where the data holder does

not trust the parties with whom he/she communicates, the proposed standard is the *Crypto Module*. The cryptographer relies on his/her Secure Multiparty Computation SMC [43] technique, with uniform cryptography. Secure multi-part computing is a cryptographic technique that allows different parties to perform calculations through inputs, while keeping those inputs private and ensuring the required levels of privacy and security. The SMC methodology consists of calculating a function based on private values held by the members of a group. The basic approach guarantees this calculation without each member disclosing its private value (input to the function). The model can also be considered as a trusted basis by using the public key structure, or by using a symbol sequence as a common reference. In general, a multi-member secure computing problem is concerned with computing any function in a distributed network. Each participant possesses one of the arguments, thus ensuring the independence of the arguments, the correctness of the calculation and that no additional information will be revealed to a participant, which can be deduced from his argument and the result of the calculation. A common strategy is to consider the reliability of service providers, or to consider the existence of a third party, which is risky in a dynamic and malicious environment.

The common feature of these problems is that two or more members want to perform a calculation based on their own private arguments, but no member wants to reveal their own argument to someone else. The problem is how such a calculation can be performed, while maintaining the privacy of the arguments. This problem is solved by the use of *homomorphic encryption*, which allows calculations to be performed on encrypted data. The uniform cryptographic system uses a public key to encrypt the data and it allows only the person with the appropriate private key to access the unencrypted data.

4.7 Federating Module

The federal learning system is the part of the proposed architecture which implements *federal learning* (FEDL) in the general operating framework of the system. Federal Learning [6, 44] is a technique based on decentralized algorithm training on multiple decentralized devices or servers that have local samples of a data set but do not exchange them. This approach contrasts with traditional central machine learning techniques, where all data sets are contained on a single server, even in decentralized approaches that assume that local data samples are distributed evenly. More generally, FEDL allows the creation of a common, robust, machine learning model without data sharing, thus addressing critical issues such as the protection and security of personal data, securing access rights and interacting with heterogeneous data or systems. Let us assume that there are N data holders $\{F_1, F_2, \ldots, F_N\}$ and everyone wants to train a machine learning model by integrating their respective data $\{D_1, D_2, \ldots, D_N\}$. A conventional method proposes to gather all the data into a single set $D_{sum} = D_1 \cup D_1 \cup \cdots \cup D_N$ in order to train a model M_{sum}.

Instead a federal learning system suggests that data holders collaboratively train a model M_{fed}, in which the data D_1 of the owner's F_1 are not exposed to the rest of the owners. Moreover, the accuracy V_{fed} of the model M_{fed}, should approach the accuracy V_{sum} of the M_{sum}. More specifically, if δ is a negative number, the federal learning algorithm has a loss of δ-accuracy, as presented in the following mathematical

expression 1:

$$\left|V_{fed} - V_{sum}\right| < \delta \tag{1}$$

The proposed *Federating Module* is based on the horizontal federal learning architecture, as presented in Fig. 6 below.

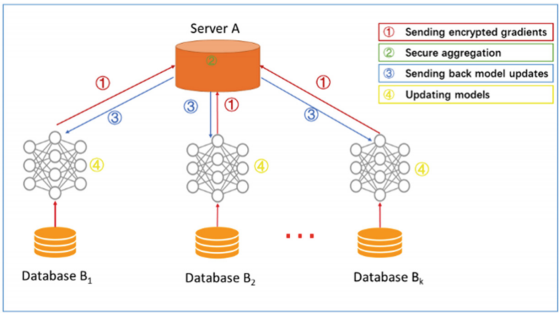

Fig. 6. Horizontal federated learning (https://medium.com/disassembly/)

In this system, participants with the same data structure, collaboratively learn a ML model, with the help of a trusted server.

The training process of the proposed system comprises of the following four steps:

1. Step 1: Participants locally calculate the gradient descent optimization steps and send them encrypted with uniform encryption via the block chain network to the host server.
2. Step 2: The server performs secure aggregation using the *Weighted Average Ensemble* technique, ensuring that information remains private.
3. Step 3: The server sends the aggregated results to the participants.
4. Step 4: Participants update their respective model.

The repetitions continue until the loss function converges, thus completing the training process.

In order to create an accurate and robust system, which is able to keep more or less stable performance with the presence of new information, the *Weighted Average Ensemble* [33, 45] technique has been used. The rationale behind this choice is related to the fact that very often, in multifactorial problems of high complexity, the results of the prediction are multivariate. This can be attributed to the sensitivity of correlational models and to the heterogeneity of data flows. The two most important advantages of *Ensemble* techniques are the following: First, they offer better predictability and more stable outcome, as the overall behavior of a multiple model is less noisy than that of a single one. Second, an *Ensemble* method can lead to very stable predictions, while offering generalization. The obtained models can also reduce the values of bias, and variance in an effort to eliminate overfitting, thus keeping robust learning capacity. Essentially a weighted set is an extension of a model that simply uses the average of

the set, since each member's contribution to the final forecast is weighted by the overall performance of the developed approach. The weights are small positive values and the overall sum is equal to one, allowing them to indicate the confidence level or the expected performance. Each data stream is controlled by every federal learning subsystem and the respective classification accuracy is extracted. The optimization steps of the *gradient descent* process are then sent on the central server, where the steps of the model with maximum accuracy receive a specific weight. Using the *dynamic weighted average*, it is significantly easier to visualize the trends of the estimated situation. This eliminates—or at least minimizes—the statistical noise of the data streams. It is one of the best ways to assess the strength of a trend and its probability of reversal. Greater weights are assigned to classification with higher accuracy, before the start of a new situation or event, thus allowing a quick and optimal decision to be made.

It is also important to note that this dynamic process ensures the adaptation of the system to new situations, by offering impartiality and generalization. This is one of the key issues in the field of ML, thus implementing a system capable of responding to highly complex problems. Finally, this architecture significantly speeds up the process of making the best decision, with the rapid convergence of the multiple model.

4.8 Blockchain Module

The blockchain module is used for the advanced security of privacy and the integrity of transactions in the Industry 4.0 network. The blockchain [8–10] architecture operates as a distributed database or global registry, which maintains logs of all transactions on a network. A transaction is a time-stamped record that specifies the identity and type of operation, the operation itself, and the users participating in it. Transactions are combined into blocks where each block is identified based on a cryptographic hash. An open public-private key pair is usually formed for each user. It is linked to their account and it is used to sign a transaction and to clearly identify ownership of a function. To form a block in the block chain, a hash function with all transaction information is computed, and then the hash value is used to calculate the hash of the block. Thus, the transaction information becomes unchanged and ensures the security, authenticity and durability of the data storage in the block chain. If there are conflicting transactions in the network, only one of them is selected to be part of the block. The blocks are added to the block chain at regular intervals to form a linear sequence where each block reports the hash of the previous block, thus forming a chain of blocks. This chain is maintained by the nodes of the network, with each node being able to execute and record all the transactions that take place. It should be noted that the trading account is designed to be distributed.

4.9 Elastic Stack Module

The widely used *Elastic Stack* system [39] containing *Elastic search*, *Logstash* and *Kibana* is used for the complete and efficient management and control of the results of the analysis. These allow the results to be stored, searched and visualized. Specifically, Logstash is an open source tool for collecting and managing used data and metrics. Similarly, Elastic search is a distributed open source search engine based on the *Apache*

Lucene library. It provides full-text search through web interface following the *Representational State Transfer* (REST) model. Finally, *Kibana* is a web application that runs exclusively on browsers without any special requirements and is used as a tool for visualizing and presenting data found in Elasticsearch. It offers high-precision and aesthetic graphs and statistical analyses that allow the interpretation of data in real time. This framework, also contains the additional web interface to facilitate management and to display the complex analysis results.

4.10 Privacy Module

As the amount of data that an organization collects and uses for analyses increases, so do concerns of privacy and security. The system uses differential privacy [46], so that independent observers looking for a specific content, or seeing an output of its data displayed by the *Elastic Stack Module*, cannot understand if the information comes from a specific system. Differential privacy is a set of systems and practices that help keep the data of individuals safe and private. In traditional scenarios, raw data is stored in files and databases. When users analyze data, they typically use the raw data. This is a concern because it might infringe on an individual's privacy. Differential privacy tries to deal with this problem by adding "noise" or randomness to the data so that users can't identify any individual data points. At the least, such a system provides plausible deniability. Therefore, the privacy of individuals is preserved with limited impact on the accuracy of the data.

In differentially private systems, data is shared through requests called queries. When a user submits a query for data, operations known as privacy method add noise to the requested data. Privacy mechanisms return an approximation of the data instead of the raw data. This privacy-preserving result appears in a report. Reports consist of two parts, the actual data computed and a description of how the data was created [47].

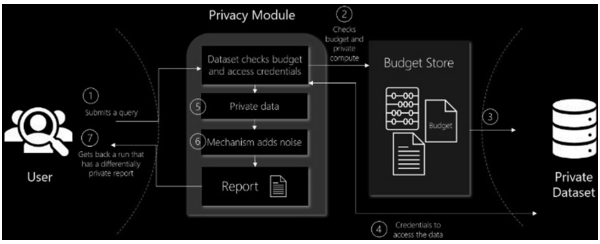

Fig. 7. Differential privacy in machine learning (https://docs.microsoft.com/)

Differential privacy tries to protect against the possibility that a user can produce an indefinite number of reports to eventually reveal sensitive data. A value known as epsilon measures how noisy or private a report is. Epsilon has an inverse relationship to noise or privacy. The lower the epsilon, the noisier (and private) the data is. Epsilon values are non-negative. Values below 1 provide full plausible deniability. Anything above 1 comes with a higher risk of exposure of the actual data. As you implement differentially private systems, you want to produce reports with epsilon values between 0 and 1. Another

value directly correlated to epsilon is delta. Delta is a measure of the probability that a report is not fully private. The higher the delta, the higher the epsilon. Because these values are correlated, epsilon is used more often.

To ensure privacy in systems where multiple queries are allowed, differential privacy defines a rate limit. This limit is known as a privacy budget. Privacy budgets prevent data from being recreated through multiple queries. Privacy budgets are allocated an epsilon amount, typically between 1 and 3 to limit the risk of reidentification. As reports are generated, privacy budgets keep track of the epsilon value of individual reports as well as the aggregate for all reports. After a privacy budget is spent or depleted, users can no longer access data. Although the preservation of privacy should be the goal, there is a tradeoff when it comes to usability and reliability of the data. In data analytics, accuracy can be thought of as a measure of uncertainty introduced by sampling errors. This uncertainty tends to fall within certain bounds. Accuracy from a differential privacy perspective instead measures the reliability of the data, which is affected by the uncertainty introduced by the privacy mechanisms. In short, a higher level of noise or privacy translates to data that has a lower epsilon, accuracy, and reliability.

5 Conclusion

This paper presents an innovative adaptive federal automatic post-machine learning architecture using distributed block chain technology for digital security and privacy under the Industry 4.0 standard. In this architecture, intelligent standards and technologies are adopted and synthesized in an innovative framework that enhances the active security methods of industrial information systems and networks, while ensuring privacy and industrial secrecy.

The proposed architecture aims to improve the security and functionality of industrial applications by providing a decentralized, reliable, encrypted network for communication between devices. In this way it contributes towards the management of critical industrial infrastructure. In essence, this architecture is called upon to fill a key gap in the way the industry operates, in the context of the convergence of heterogeneous infrastructures and financial investments in the medium to long term. The features of the proposed architecture allow the analysis, forecasting, monitoring and management of complex situations related to information systems security, optimally combining and implementing a hybrid system with the most technologically advanced methods of computational intelligence.

The idea of standardizing a novel and robust architecture has emerged based on the application of a single, universal method capable to cover all the industrial requirements of the new era. It combines the most up-to-date methods and it is able to complete specialized processes for the development of modern information systems' security applications. This is achieved through a customizable, flexible and easy-to-use interface. The design and development of this architecture, is based on the use of the most modern software development methods, employing adaptive artificial intelligence algorithms. Overall, the architecture is based on a multi-layered, modular model. It is following a distributed design, which allows the connection to any existing systems, as well as the expansion through the integration of new functions. The main features of the proposed approach which can offer new functions and perspectives to industrial production,

are Quality, Support, Reliability, Bidirectional Communication, Security, Privacy and Decentralization of services. Attempting an in-depth analysis of how this architecture works, it is obvious that the most basic decentralized industrial applications that can benefit from its use are On-Demand Manufacturing, Smart Diagnostics and Machine Maintenance, Traceability, Product Certification, Predictive Manufacturing and Tracking Supplier's Identity & Reputation.

Given that so far no such integrated architectures have been proposed that can comprehensively support the security and protection of industrial confidentiality, or even realistically approach the complexity of this environment, a thorough comparison cannot be performed. Future research directions that could expand and promote the proposed architecture, mainly concern the possibilities of optimization and additional integration of services or combination between the used ones, which will further expand the dynamics of the architecture. Respectively, a future development that is considered particularly important for the strengthening of the forecasting system is the integration of methods that can stage the uncertainty and the non-linear ways of development of modern digital threats. In addition, it would be of particular interest to integrate the Neural Search Module, with algorithmic approaches that lead to the dynamic rearrangement of automatic meta-models that update their predictive power in real time, based on the evolution of a situation. Thus, the system will create real-time dynamic self-identifying neural networks (online learning). This would be particularly useful in evaluation situations of large-scale data, such as in the cases of industrial ecosystems' digital security. Finally, the new architectural formulations that accompany Industry 4.0, should be studied in order to identify gaps or omissions in the proposed architecture. This will enable modelling of new security and industrial privacy violation threats and it will lead to the design of the appropriate countermeasures.

References

1. Kannengiesser, U., Muller, H.: Towards viewpoint-oriented engineering for Industry 4.0: a standards-based approach. In: 2018 IEEE Industrial Cyber-Physical Systems (ICPS), St. Petersburg, pp. 51–56 (2018). https://doi.org/10.1109/ICPHYS.2018.8387636
2. Hossain, Md.M., Fotouhi, M., Hasan, R.: Towards an analysis of security issues, challenges, and open problems in the Internet of Things. In: 2015 IEEE World Congress on Services, NY, USA, pp. 21–28 (2015). https://doi.org/10.1109/SERVICES.2015.12
3. Banafa, A.: 2 The Industrial Internet of Things (IIoT): challenges, requirements and benefits. In: Secure and Smart Internet of Things (IoT): Using Blockchain and AI, pp. 7–12. River Publishers (2018)
4. Ankele, R., Marksteiner, S., Nahrgang, K., Vallant, H.: Requirements and recommendations for IoT/IIoT models to automate security assurance through threat modelling, security analysis and penetration testing. In: Proceedings of the 14th International Conference on Availability, Reliability and Security, NY, USA, pp. 1–8 (2019). https://doi.org/10.1145/3339252.3341482
5. Li, J.-Q., Yu, F.R., Deng, G., Luo, C., Ming, Z., Yan, Q.: Industrial internet: a survey on the enabling technologies, applications, and challenges. IEEE Commun. Surv. Tutor. **19**(3), 1504–1526 (2017). https://doi.org/10.1109/COMST.2017.2691349
6. Wahab, O.A., Mourad, A., Otrok, H., Taleb, T.: Federated machine learning: survey, multi-level classification, desirable criteria and future directions in communication and networking systems. IEEE Commun. Surv. Tutor., 1 (2021). https://doi.org/10.1109/COMST.2021.3058573

7. Gebremichael, T., et al.: Security and privacy in the Industrial Internet of Things: current standards and future challenges. IEEE Access **8**, 152351–152366 (2020). https://doi.org/10.1109/ACCESS.2020.3016937
8. Demertzis, K.: Blockchained federated learning for threat defense. arXiv:2102.12746 Cs, February 2021. Accessed 26 Feb 2021
9. Rantos, K., Drosatos, G., Demertzis, K., Ilioudis, C., Papanikolaou, A.: Blockchain-based consents management for personal data processing in the IoT ecosystem, pp. 572–577, February 2021. https://www.scitepress.org/PublicationsDetail.aspx?ID=+u1w9%2fItJqY%3d&t=1. Accessed 16 Feb 2021
10. Rantos, K., Drosatos, G., Demertzis, K., Ilioudis, C., Papanikolaou, A., Kritsas, A.: ADvoCATE: a consent management platform for personal data processing in the IoT using blockchain technology. In: Lanet, J.-L., Toma, C. (eds.) SECITC 2018. LNCS, vol. 11359, pp. 300–313. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-12942-2_23
11. Demertzis, K., Iliadis, L., Tziritas, N., Kikiras, P.: Anomaly detection via blockchained deep learning smart contracts in industry 4.0. Neural Comput. Appl. **32**(23), 17361–17378 (2020). https://doi.org/10.1007/s00521-020-05189-8
12. Cook, A.A., Mısırlı, G., Fan, Z.: Anomaly detection for IoT time-series data: a survey. IEEE Internet Things J. **7**(7), 6481–6494 (2020). https://doi.org/10.1109/JIOT.2019.2958185
13. Demertzis, K., Iliadis, L.: A hybrid network anomaly and intrusion detection approach based on evolving spiking neural network classification. In: Sideridis, A.B., Kardasiadou, Z., Yialouris, C.P., Zorkadis, V. (eds.) E-Democracy 2013. CCIS, vol. 441, pp. 11–23. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11710-2_2
14. Basyoni, L., Fetais, N., Erbad, A., Mohamed, A., Guizani, M.: Traffic analysis attacks on Tor: a survey. In: 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), pp. 183–188 (2020). https://doi.org/10.1109/ICIoT48696.2020.9089497
15. Alhawi, O.M.K., Baldwin, J., Dehghantanha, A.: Leveraging machine learning techniques for windows ransomware network traffic detection. In: Dehghantanha, A., Conti, M., Dargahi, T. (eds.) Cyber Threat Intelligence. AIS, vol. 70, pp. 93–106. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-73951-9_5
16. Balabanova, I., Kostadinova, S., Markova, V., Georgiev, G.: Analysis and categorization of traffic streams by artificial intelligence. In: 2019 International Conference on Biomedical Innovations and Applications (BIA), pp. 1–5 (2019). https://doi.org/10.1109/BIA48344.2019.8967475
17. Demertzis, K., Iliadis, L., Bougoudis, I.: Gryphon: a semi-supervised anomaly detection system based on one-class evolving spiking neural network. Neural Comput. Appl. **32**(9), 4303–4314 (2019). https://doi.org/10.1007/s00521-019-04363-x
18. Hsu, C.-H., Huang, C.-Y., Chen, K.-T.: Fast-flux bot detection in real time. In: Jha, S., Sommer, R., Kreibich, C. (eds.) RAID 2010. LNCS, vol. 6307, pp. 464–483. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15512-3_24
19. Chen, Y.-S., Chen, Y.-M.: Combining incremental hidden Markov model and adaboost algorithm for anomaly intrusion detection. In: Proceedings of the ACM SIGKDD Workshop on CyberSecurity and Intelligence Informatics, New York, NY, USA, pp. 3–9 (2009). https://doi.org/10.1145/1599272.1599276
20. Alshammari, R., Nur Zincir-Heywood, A.: Identification of VoIP encrypted traffic using a machine learning approach. J. King Saud Univ. Comput. Inf. Sci. **27**(1), 77–92 (2015). https://doi.org/10.1016/j.jksuci.2014.03.013
21. Demertzis, K., Tziritas, N., Kikiras, P., Sanchez, S.L., Iliadis, L.: The next generation cognitive security operations center: adaptive analytic lambda architecture for efficient defense against adversarial attacks. Big Data Cogn. Comput. **3**(1), 6 (2019). https://doi.org/10.3390/bdcc3010006

22. Lee, H., Veeraraghavan, M., Li, H., Chong, E.K.P.: Lambda scheduling algorithm for file transfers on high-speed optical circuits. In: IEEE International Symposium on Cluster Computing and the Grid, CCGrid 2004, pp. 617–624 (2004). https://doi.org/10.1109/CCGrid.2004.1336671
23. Chen, L., Li, T., Abdulhayoglu, M., Ye, Y.: Intelligent malware detection based on file relation graphs. In: Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015), pp. 85–92 (2015). https://doi.org/10.1109/ICOSC.2015.7050784
24. Sun, Y., Wang, Z., Liu, H., Du, C., Yuan, J.: Online ensemble using adaptive windowing for data streams with concept drift. Int. J. Distrib. Sens. Netw. **12**(5), 4218973 (2016). https://doi.org/10.1155/2016/4218973
25. Sobhani, P., Beigy, H.: New drift detection method for data streams. In: Bouchachia, A. (ed.) ICAIS 2011. LNCS (LNAI), vol. 6943, pp. 88–97. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23857-4_12
26. Chen, Q., Abdelwahed, S.: A model-based approach to self-protection in SCADA systems. Presented at the 9th International Workshop on Feedback Computing (Feedback Computing 2014) (2014). https://www.usenix.org/conference/feedbackcomputing14/workshop-program/presentation/chen. Accessed 29 Mar 2021
27. Soupionis, Y., Benoist, T.: Cyber attacks in power grid ICT systems leading to financial disturbance. In: Panayiotou, C.G.G., Ellinas, G., Kyriakides, E., Polycarpou, M.M.M. (eds.) CRITIS 2014. LNCS, vol. 8985, pp. 256–267. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-31664-2_26
28. Demertzis, K., Rantos, K., Drosatos, G.: A dynamic intelligent policies analysis mechanism for personal data processing in the IoT ecosystem. Big Data Cogn. Comput. **4**(2), 9 (2020). https://doi.org/10.3390/bdcc4020009
29. Sikeridis, D., Bidram, A., Devetsikiotis, M., Reno, M.J.: A blockchain-based mechanism for secure data exchange in smart grid protection systems. In: 2020 IEEE 17th Annual Consumer Communications Networking Conference (CCNC), pp. 1–6 (2020). https://doi.org/10.1109/CCNC46108.2020.9045368
30. Llopis, S., et al.: A comparative analysis of visualisation techniques to achieve cyber situational awareness in the military. In: 2018 International Conference on Military Communications and Information Systems (ICMCIS), pp. 1–7 (2018). https://doi.org/10.1109/ICMCIS.2018.8398693
31. Çınar, C., Alkan, M., Dörterler, M., Doğru, İ.A.: A study on advanced persistent threat. In: 2018 3rd International Conference on Computer Science and Engineering (UBMK), pp. 116–121 (2018). https://doi.org/10.1109/UBMK.2018.8566348
32. Azzedin, F., Suwad, H., Alyafeai, Z.: Counter measuring zero day attacks: asset-based approach. In: 2017 International Conference on High Performance Computing Simulation (HPCS), pp. 854–857 (2017). https://doi.org/10.1109/HPCS.2017.129
33. Demertzis, K., Iliadis, L., Anezakis, V.-D.: A dynamic ensemble learning framework for data stream analysis and real-time threat detection. In: Kůrková, V., Manolopoulos, Y., Hammer, B., Iliadis, L., Maglogiannis, I. (eds.) ICANN 2018. LNCS, vol. 11139, pp. 669–681. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01418-6_66
34. Jirsik, T., Cermak, M., Tovarnak, D., Celeda, P.: Toward stream-based IP flow analysis. IEEE Commun. Mag. **55**(7), 70–76 (2017). https://doi.org/10.1109/MCOM.2017.1600972
35. Čermák, M., Tovarňák, D., Laštovička, M., Čeleda, P.: A performance benchmark for NetFlow data analysis on distributed stream processing systems. In: NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium, pp. 919–924 (2016). https://doi.org/10.1109/NOMS.2016.7502926
36. CESNET/ipfixcol2: CESNET (2021)

37. Apache Kafka: Apache Kafka. https://kafka.apache.org/. Accessed 29 Mar 2021
38. Apache Spark^{TM} - Unified Analytics Engine for Big Data. http://spark.apache.org/. Accessed 29 Mar 2021
39. Free and Open Search: The Creators of Elasticsearch. ELK & Kibana | Elastic. https://www.elastic.co/. Accessed 29 Mar 2021
40. Elsken, T., Metzen, J.H., Hutter, F.: Neural architecture search: a survey. J. Mach. Learn. Res. **20**(55), 1–21 (2019)
41. Jin, H., Song, Q., Hu, X.: Auto-Keras: an efficient neural architecture search system. arXiv: 1806.10282 Cs Stat (2019). Accessed 29 Mar 2021
42. Amazon Kinesis - Process & Analyze Streaming Data - Amazon Web Services. Amazon Web Services, Inc. https://aws.amazon.com/kinesis/. Accessed 29 Mar 2021
43. Lindell, Y.: Secure multiparty computation (MPC), 300 (2020). http://eprint.iacr.org/2020/300. Accessed 29 Mar 2021
44. Korkmaz, C., Kocas, H.E., Uysal, A., Masry, A., Ozkasap, O., Akgun, B.: Chain FL: decentralized federated machine learning via blockchain. In: 2020 Second International Conference on Blockchain Computing and Applications (BCCA), pp. 140–146 (2020). https://doi.org/10.1109/BCCA50787.2020.9274451
45. Demertzis, K., Iliadis, L., Anezakis, V.: MOLESTRA: a multi-task learning approach for real-time big data analytics. In: 2018 Innovations in Intelligent Systems and Applications (INISTA), pp. 1–8 (2018). https://doi.org/10.1109/INISTA.2018.8466306
46. Dwork, C.: Differential privacy. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4052, pp. 1–12. Springer, Heidelberg (2006). https://doi.org/10.1007/11787006_1
47. luisquintanilla: Differential privacy in machine learning (preview) - Azure Machine Learning. https://docs.microsoft.com/en-us/azure/machine-learning/concept-differential-privacy. Accessed 29 Mar 2021