Blockchain Enabled Edge-Fog and Cloud Based Architecture for IoT

Aravendra Kumar Sharma (Dept. of Computer Science & Engineering), Researcher, SunRise University, Alwar (Raj.)
Dr. Kamal Kumar Srivastava, Professor (Dept. of Computer Science & Engineering), SunRise University, Alwar (Raj.)

ABSTRACT

This paper addresses critical challenges in current blockchain-enabled Internet of Things (IoT) architectures, identifying key issues that warrant architectural modifications. The proposed solution, termed the "Blockchain-Enabled Edge-Fog and Cloud-Based Architecture for IoTs," is introduced in subsequent sections to overcome these challenges. The Issues section highlights various problems within existing architectures, setting the stage for the innovative solution presented in this work. It delves into a comprehensive exploration of current IoT architectural solutions, providing a basis for contrasting and comparing them with the proposed architecture. The novel approach integrates edge, fog, and cloud computing to create a robust and efficient framework for IoT systems, demonstrating its potential to address the identified issues and propel blockchain-enabled IoTs towards enhanced scalability, security, and performance.

Keywords: Internet of Things, Blockchain-Enabled Edge-Fog and Cloud-Based Architecture

1.1 Introduction

To get around the problems with relying on a single party or failing to back up data, the centralised IoV system uses blockchain technology. In most cases, where the vehicles' identities need to remain secret, the anonymity offered by the blockchain works perfectly. Multiple miners working together to process a single transaction is the foundation of the blockchain. Although it guarantees robustness, it necessitates the use of numerous miners to finish a single transaction. Therefore, a number of important obstacles must be overcome before blockchain may be used in the IoV. Mining for blocks usually requires a lot of processing power and isn't good for low latency. As more vehicles join the network, its complexity rises. The amount of traffic that blockchain generates as a whole is increased. If we want to create the best model for blockchain integration with IoTs, we need to alter our architecture. To begin, this article identifies a few problems with blockchained IoTs that necessitate new architectural adjustments. Next, we will go over the proposed design, which is called the blockchain-enabled edge-fog and cloud-based architecture for the Internet of Things.

1.1.1 Existing Internet of Things Architectural Solutions

There has been a lot of progress in the Internet of Things (IoT) technology business in recent years, and now is the time to examine those architecturally obvious subjects. Think about how the Internet of Things (IoT) stack as a whole consists of multiple sub-systems; this makes it hard to figure out where to start and how to show the system's architectural framework in the end. There appears to be a common function for much of the available architecture. It should be kept in mind, though, that no universally accepted IoT architecture exists at this time, hence there is no worldwide solution.

1.1.1.1 Three-Level Architecture

Developed and implemented on a variety of structures, the three-level architecture is the backbone of the Internet of Things (IoT). As shown in Figure 1.1, the three levels of the Internet of Things (IoT) architecture are physical, network, and application, with sensing, transport, and implementation occurring at each level.1. Gupta et al. (2020) and Cirani et al. (2014) presented a framework for a large-scale IoT network that is both self-configuring and flexible. Information, internet-based, and sensors and actuators are the three main parts of the Internet of Things (IoT). Internet of Things (IoT) deployment is generally in line with modern society, wherein things and people are essentially linked through wireless sensors (Tsai et al., 2014). The Internet of Things is commonly described as having three stages: vision, network, and implementation (Mahmoud et al., 2015). When describing the architecture of the Internet of Things, the sensor layer—sometimes called the "perception layer"—is positioned as the foundational layer. The Internet of Things (IoT) architecture typically features networking, sometimes called transmission, as an intermediate layer. The Internet of Things architecture is topped by the "enterprise layer," also called the framework layer (Al-Fuqaha et al., 2015). Things like "smart grids," "smart towns," and "smart mobility" are good examples.

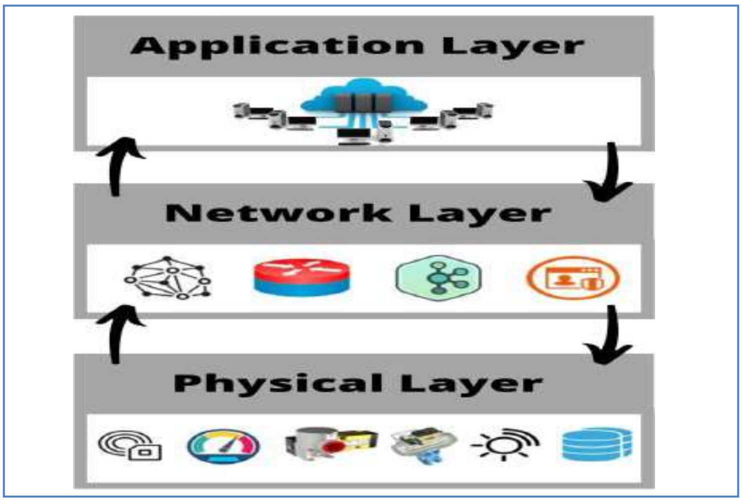

Fig. 1.1 Three Layer Architecture of IoT

The physical or perceptual Internet of Things (IoT) sheet is comprised of devices such as sensors, actuators, and other smart instruments. These are called "things" on the Internet of Things. In instance, devices typically link to the cloud by wired or regional RF networks. This is often accomplished with the use of gateways. The computers that make up the Internet of Things are known as "edge nodes" because of their central location within the network. System sensitivity is affected by factors such as low latency, RF protection, strength, robustness, and strength, even in I/O protocols and RF interfaces. In addition to RF protection, strength, resilience, and general machine sensitivity, low latency is an important factor in I/O protocols and RF interfaces. Determining the level of computer stability required in the design is of paramount importance.

A number of contemporary equipment necessitate IoT programming for ever-increasingly complex customisation. In the "IoT" period prior to this, many actuators and older machines rely on serial or analogue interfaces. Microcontrollers, systems on modules (SOMs), or single-board computers (such as the Raspberry Pi) are common components of such machines. One possible feature of such a collector is the ability to link the main gateway or additional keymaster and/or resource nodes to the central hub. Plus, there are some situations where they can act as a portal. IoT When connecting smart devices to cloud services, gateways play a crucial role as an intermediate. Virtualization is common for edge sensors that run on software or hardware in the real world. To support a high density of edge nodes, large-scale IoT systems can employ several gateways. Among their many possible functions, data normalisation, connectivity, and cloud-to-physical-system layer conversion are among the most important. The data is transferred from the cloud to the physical system layer through a gateway. Some people call IoT gateways "power tiers" or intelligent gateways. Telemetry, protocol conversion, AI, pre-processing of massive sensor data sets, and the provisioning of a plethora of highly personalised devices are just a few examples of the computing and other peripheral capabilities enabled by modern gateways. Data protection and surveillance on intelligent gateways are now considered standard practice to prevent disruptive man-in-the-middle attacks on IoT networks. Some gateways, like NetBurner, provide an integrated operating system that is tailored to the Internet of Things (IoT) and also offer low-level compatibility for other interfaces. Management of interfaces, memory, and input/output is not an easy operation. According to Google Cloud, there aren't enough abstractions for all the sensors and actuators that could be encountered when building Internet of Things solutions. To make libraries accessible, standard protocols are typically employed. When it comes to software and hardware protocols, NetBurner has you covered with specialised libraries and commercial development kits (SDKs) that often contain the most powerful and well-integrated libraries. For data, the cloud is the place to be. It uses either a wired or wireless connection to link up with the gateway. Web services, data centres, or even whole services can exist in the cloud, thanks to tools like Amazon Web Services (AWS) and Google Cloud Storage. Strong Internet of Things (IoT) applications can run smoothly on it and have enough space for data filtering, in-depth data analysis, reporting, third-party APIs, tracking, and user interfaces. For three-layer Internet of Things (IoT) systems with linked edge devices (such as gateways), the cloud architecture governs system configurations, automation, and event trigger controls.

1.1.1.2: Architecture Based on Quality of Service

Incorporating Quality of Service (QoS) standards into smart city and other urban system applications was suggested by Zhang et al. (2018) using the three-layer design shown in Fig.1.2. Potential network implementations that enhance Internet access include application-specific overlays, which alleviate the load and latency that nodes encounter in the disconnected ecosystem when using the Internet.

Applications Layer: The Quality of Service (QoS) provided directly to end users by applications and service layers has become the gold standard for assessing the QoS of the Internet of Things (IoT). Here are the main features:

The Internet of Things (IoT) will offer applications in the here and now, much like traditional networks used to offer services at various points in their lifetimes. However, it is important to assign the duties implied by the service.

Delay in Service: The response time-to-service (RTSP) metric measures how long it takes for a user to input data and how long it takes for the application to return the results. It is a well-known criterion for distinguishing the baseline performance.

Precision of Service: One key differentiator in the era of Internet of Things applications should be the precision of service they provide. This includes developing apps that accurately execute different customer order requests.

Loading Service: This data processing capacity discusses the possibilities of these IoT apps for various resources or the quantity of concurrent users.

Priority Service: Priority service is a way for the requestee to specify the level of service they need and the degree to which it will be provided. Firstly, it is an Internet of Things (IoT) service with a unique selling point.

We have the flexibility to modify conditions for each service level or assign unique permissions to different user positions, locations, times, or events within the same service level. The operational requirements of dispersed Internet of Things devices can vary. While these traits are relevant to quality of service, they are not limited to it. Data confidentiality, mobility, authentication, encryption, transmission latency, packet error rate, location data, terminal availability (market dependent), and packet error rate are some of these features. In order to manage their unique forms, networks can make use of a wide variety of network-specific quality-of-service indicators and procedures (Network Layer). They offer varying degrees of service, be it for mobile phone networks, internet access, or something else entirely. Transmission rate, error rate, and packet latency are some quality of service indicators that quantify this transmission mechanism. However, converting the requirements from the top layer to guarantee network QoS is the toughest issue.

Perception Layer: The QoS perception layer ensures consistent monitoring by coordinating sampling, coverage, time synchronisation, and location/mobility.

Criteria of Sampling: In terms of sampling criteria, the "samples" stand for both static and dynamic descriptions of the method's adaptability. The characteristics for sensor and multimedia info include things like transfer receiving rate, sampling precision, and sample spectrum for sensors.

Coverage: This is the range of fields where the perception layer works well. The coverage ratio or the redundant coverage ratio might be used to determine it.

Time Synchronisation: All devices used to sense at the perception level are anticipated to maintain a uniform time relationship in order to appropriately sample the outcomes that are time-dependent.

Mobility and Positioning: In some implementations of the perception layer of the Internet of Things (IoT), such as vehicle tracking systems, additional mobility information is needed. This is because the physical layer of these systems often contains sensors that can move from one place to another. Therefore, in order to get reliable results, it is necessary to employ the best position measurement techniques.

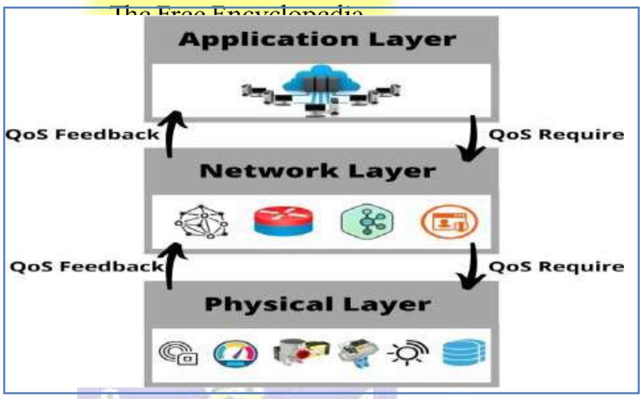

Fig. 1.2 QoS Based

Architecture in IoT

1.1.2 . IoT Blockchain Architecture:

According to this design (Hang et al., 2019), as illustrated in Fig. 1.3, developers can easily swap out or add new modules to one layer without affecting the others.

1.1.2.1. Physical Layer: The physical layer of the internet of things (IoT) is made up of numerous interconnected systems that may store data, communicate with one another, and use computers.

1.1.2.2. Communication Layer : Since physical devices do not have access to global internet protocols (IPs), network self-organization relies on the installation, maintenance, and routing of network protocols, which are primarily performed by the communication layer . Message broker administration, network and protection control, and other service offering modules are also available.

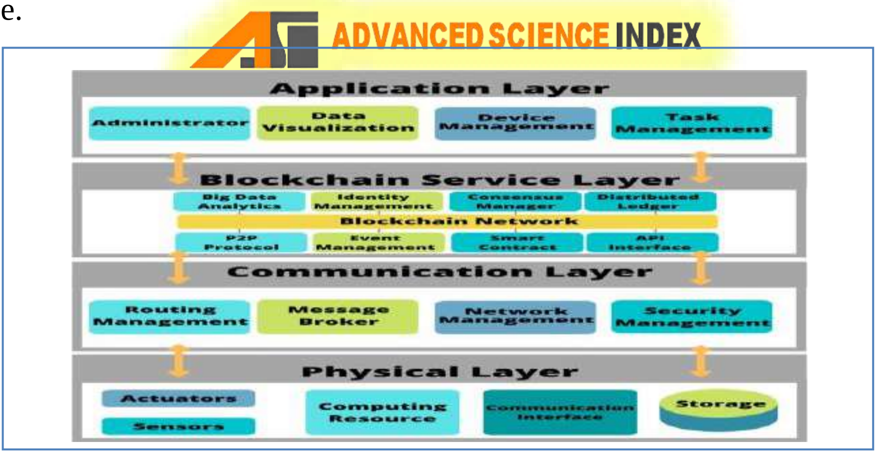

Fig. 1.3 Blockchain Based IoT Architecture

1.1.3 The Blockchain Service Layer: The Internet of Things (IoT) blockchain service layer incorporates several blockchain capabilities, such as identity management, consensus, and peer-to-peer (P2P) networking, among others, and comprises all the modules that handle key resources. Each user has the exact same version of the records in the network-wide ledger because of the synchronisation, mirroring, and exchange of records. Not only that, but it also remembers configuration details and information about the physical sensors. Any copy of the ledger will reflect a single update within minutes, or even seconds, in certain cases. By allowing only validated members to execute P2P transactions, the ledger might be used as a permissioning mechanism for non-members. Big data and data mining enhance the blockchain's processing and storage capabilities. Hierarchical ledger formats are ideal for numerous parties' independent contributions since they store a lot of transaction records from different sources. Everyone has access to the same pool of information through a single network. The use of smart contracts by an external programme controls the addition and change of ledger entries. Even though it's common on most networks, you can usually find it on any peer. An event is sent whenever the event management system receives a new block or when a condition in the smart contract is satisfied. The application programming interface (API) acquaints clients with the network's blockchain resources so that they can use and manage the blockchain.

Application Layer: At the very top is the application layer, which is responsible for displaying various types of data in a graphical and command format. This data is subsequently utilised for the purpose of monitoring and managing physical objects.

1.2 Related Literature

Year: 2018

Author: Sharma and Jain

Related Work: "A Survey on Fog and Edge Computing: Architectures, Applications, and Research Directions"

Conclusion: Sharma and Jain provide a comprehensive survey on fog and edge computing, exploring various architectures and applications. Their conclusion emphasizes the need for optimized resource allocation, efficient communication protocols, and seamless integration with cloud services to enhance the performance and reliability of IoT systems in the context of fog and edge computing.

Year: 2019

Author: Sun et al.

Related Work: "Blockchain-Based Secure Firmware Update for Internet of Things Devices"

Conclusion: The authors propose a blockchain-based solution to secure firmware updates for IoT devices. They argue that blockchain can enhance security and integrity in IoT systems by providing a decentralized and tamper-proof ledger for managing firmware updates. Their conclusion suggests that integrating blockchain technology can address security concerns in IoT ecosystems effectively.

Year: 2019

Author: Khan et al.

Related Work: "Cloud Computing and Internet of Things: A Survey"

Conclusion: Khan et al. present a survey on the integration of cloud computing and IoT. They discuss various challenges, including data security, scalability, and interoperability. The conclusion highlights the importance of developing efficient communication protocols, edge intelligence, and collaborative solutions to optimize the synergy between cloud computing and IoT in the context of Indian scenarios.

Year: 2020

Author: Reddy et al.

Related Work: "Secure Data Sharing in Cloud-Assisted Internet of Things using Blockchain"

Conclusion: Reddy et al. focus on secure data sharing in IoT environments with the assistance of cloud computing and blockchain. Their work highlights the potential of blockchain to establish trust and security in data sharing among IoT devices. The conclusion suggests that integrating blockchain with cloud-based architectures can significantly enhance the overall security and privacy of IoT systems.

Year: 2020

Author: Singh et al.

Related Work: "Fog Computing and Its Role in the Internet of Things: A Review"

Conclusion: Singh et al. provide an overview of fog computing and its significance in IoT environments. They highlight the potential of fog computing to reduce latency, improve scalability, and enhance privacy in IoT systems. Their conclusion emphasizes the need for efficient resource management and integration with cloud and edge computing to optimize IoT operations.

Year: 2021

Author: Patel et al.

Related Work: "A Comprehensive Review on Blockchain in Edge and Fog Computing"

Conclusion: Patel et al. conduct a detailed review of the application of blockchain in edge and fog computing environments. They discuss the potential of blockchain to address security, privacy, and trust issues in these architectures. Their conclusion emphasizes the need for standardized protocols, energy-efficient consensus mechanisms, and scalability to make blockchain a viable solution for securing edge and fog computing in IoT scenarios.

Year: 2021

Author: Gupta et al.

Related Work: "Blockchain and IoT Integration: Current Status, Open Issues, and Future Directions"

Conclusion: Gupta et al. explore the integration of blockchain and IoT technologies, discussing current challenges and future directions. They argue that blockchain can enhance security, transparency, and data integrity in IoT applications. Their conclusion highlights the need for standardized protocols, interoperability, and scalability to realize the full potential of blockchain-enabled IoT architectures.

Year: 2022

Author: Kumar et al.

Related Work: "Edge Computing in Internet of Things: A Review"

Conclusion: Kumar et al. conduct a comprehensive review of edge computing in IoT deployments. They discuss various edge computing architectures, challenges, and emerging trends. Their conclusion emphasizes the importance of edge computing in reducing network congestion, enhancing real-time processing, and improving data privacy in IoT systems.

1.3. Issues

A number of factors highlight the critical nature of the IoT reference design. Here are some of them:

- Since the number of Internet of Things (IoT) devices is quickly growing and now exceeds one billion, scalability architecture is necessary. Additionally, these systems often connect continuously, therefore we require an easily accessible solution that enables deployment through data centres to provide disaster recovery.
- Personal data is captured and analysed constantly using IoT devices. Therefore, a standard for identity and access control of Internet of Things (IoT) devices and the data they publish and consume is essential.
- Internet of Things (IoT) devices are intrinsically linked; thus, a means of communication with them is necessary, regardless of obstacles such as firewalls or the conversion of network addresses.

- We must adopt regulated and automatic upgrades and enable remote control of these devices because the majority of them lack a user interface and are designed for "everyday" use.

A solution that makes system and application integration easier should be the goal of any good architecture. The limited resources and compact form factors of IoT devices dictate a number of fundamental IoT parameters that are unique to these devices. For instance, a number of requirements stem from the fact that IoT devices have limited form factors and resources. Additional requirements are derived from the production and operation of IoT devices. The procedures don't resemble modern Internet apps at all; rather, they resemble ideas for traditional digital products. It is essential to be familiar with and take into account a set of common best practices for server-side and Internet communication. Some of the most important requirements that call for different types of categories are listed below.

1.3.1 Device Management

Since many Internet of Things devices are not always being used, this isn't necessarily a good thing. It is probable and advantageous that active control will become increasingly prevalent for Internet of Things (IoT) devices, similar to how it has for personal computers, mobile phones, and other devices. It has specific requirements that are ideal for handling Internet of Things devices, such as the capacity to update firmware, remove a compromised or malicious unit, improve access control, locate a misplaced device, remotely enable or disable hardware features, remotely re-configure Wi-Fi, GPRS, or network settings, and so on.

1.3.2. Security

A key component of the IoT is security. By virtue of existing, IoT applications also collect extremely private data and bring the physical world online and vice versa. It classifies risks into three categories: those that are special to the Internet of Things (IoT) and its devices, those that are inherent to all Internet devices but were unknown to the designers of IoT devices, and those that are safety-related to prevent harm from actuator misuse.

1.3.3 Information Collecting, Analysis, and Processing

While some Internet of Things devices do include user interfaces, the vast majority of these devices depend on sensors, actuators, or a mix of the two. We are required by the system to collect data from an enormous number of devices, store it, analyse it, and then use it in some way. Reference architecture is built to support incredibly large numbers of devices. These systems generate massive amounts of data if they are continuous data sources.

1.3.4. Scalability

Any server-side design worth its salt should be infinitely scalable, able to take in data from millions of devices streaming in real time. However, there is usually a cost—in terms of complexity, software, and hardware—to a highly scalable architecture. Encouragement of scalability from small installations to a high number of devices is crucial for this design. A cloud infrastructure must have efficient deployment and scalability. To make this architecture affordable for both small and big deployments, the ability to scale out the server side onto small, cheap servers is a crucial prerequisite.

1.3.5. Communications and Connectivity

Several uses have given modern protocols like HTTP a physical home. Basic GET and POST requests can be generated by an 8-bit controller, and HTTP provides necessary centralised (and uniform) networking. The overhead of HTTP and other traditional Internet protocols might be problematic for two main reasons. Next, for smaller devices, programme memory capacity can be an issue. But the electricity requirements are the bigger worry. To satisfy such constraints, a tiny, binary protocol is needed. Additionally, there are devices that link up with gateways and those that link up directly. A gateway protocol and a cloud protocol may be necessary for devices to connect through a gateway. Lastly, transport and protocol bridging must be supported by the architecture. To keep tabs on the interface we make available to outside parties, we might implement an HTTP-based API, for instance, while still providing the system with a binary protocol.

What we have established as a set of reference architecture specs is now complete. There will be supplementary requirements for each given design. Some of those can already be met by the architecture, while others could require the inclusion of modules.

1.4 Proposed Architecture Solution

Internet of Things (IoT) designs with three to seven levels have been suggested in different research. The integration of blockchain, cloud computing, and fog/edge computing into a unified architecture is an open question. Blockchain Enabled Edge-Fog and Cloud Based Architecture for the Internet of Things (BEFC-IoT) is a novel architectural perspective proposed in this study. The foundation of this system is the ability to manage resources and ensure the security of data in transit between Internet of Things devices. A three-layer architecture consisting of a local blockchain layer, a fog layer, and an edge layer could achieve this goal. Additionally, a cloud storage layer would be added between the application layer and the network layer, and a security layer enabled by blockchain technology could be placed on top of the application layer.

1.4.1. Internet of Things Architecture—Blockchain Enabled Edge-Fog and Cloud Based

The necessity for IoT devices to store and process data has grown in importance as the number of IoT devices continues to skyrocket from millions to billions worldwide. When it comes to the data created by devices, big data is essential for collecting, processing, and making sense of all that data. The Internet of Things (IoT) presents challenges with massive sensing and controlling data sets, which calls for big data solutions. Data generated by IoT applications is often unstructured, necessitating further processing to extract useful information. Internet of Things (IoT) and big data technologies are so anticipated to offer substantial advantages. Big data storage, retrieval, and analytics will be impacted as the Internet of Things (IoT) develops into the next technology revolution. The Internet of Things would rely on constant data flow. The proliferation of IoT devices and apps would need the construction of additional data centres. Moving the data to the cloud through Platform as a Service is one such solution. The likelihood that the device's internal data could be pertinent should be considered by a company as it chooses a system for managing big data analytics. The usage of Hadoop and Hive allows for the control of extremely large amounts of data.

Apache Kafka can grow to real-time applications and is also useful for processing big data sets. Internet of Things also impacts the safety of massive data (Zhou et al., 2015). To render it useless, security at the device level of the Internet of Things is of utmost importance. Even though blockchain is the best solution for Internet of Things security, it's difficult to keep track of the enormous amounts of data required by blockchain on the local level of IoT devices. The computing and data storage technology known as "the cloud" is state-of-the-art, reliable, and capable. The addition of cloud computing to the Internet of Things, however, will bring additional challenges. There is an enormous amount of data that needs to be gathered, processed, and assessed from things like smartphones, home appliances, sensors, and IoT sensor systems. By bringing cloud computing closer to the point of need, fog/edge computing can fill the gap (Sehgal et al., 2015). Instead of putting compute and storage in the network's core, they can be dispersed throughout the fog and edge. Any network computer with the ability to retrieve data, perform computations, and communicate over the network might be considered a fog or edge computing node, as illustrated in Figure 1.4.

Figure 1.4 shows that the BEFC-IoT Architecture may be simplified into five tiers. According to the earlier conceptions of the architecture, the physical layer is the first and contains all of the elements that make up the Internet of elements. The second layer, known as the network layer, has three sub-layers. Edge computing networks are at the very bottom of the stack, with local blockchain layers and fog computing networks following closely behind. If an application needs data from the fog network, it can move it to the cloud, where it can be accessed and viewed by the application layer. The worldwide blockchain network subsequently envelops all the applications interacting with the network in a protective shield.

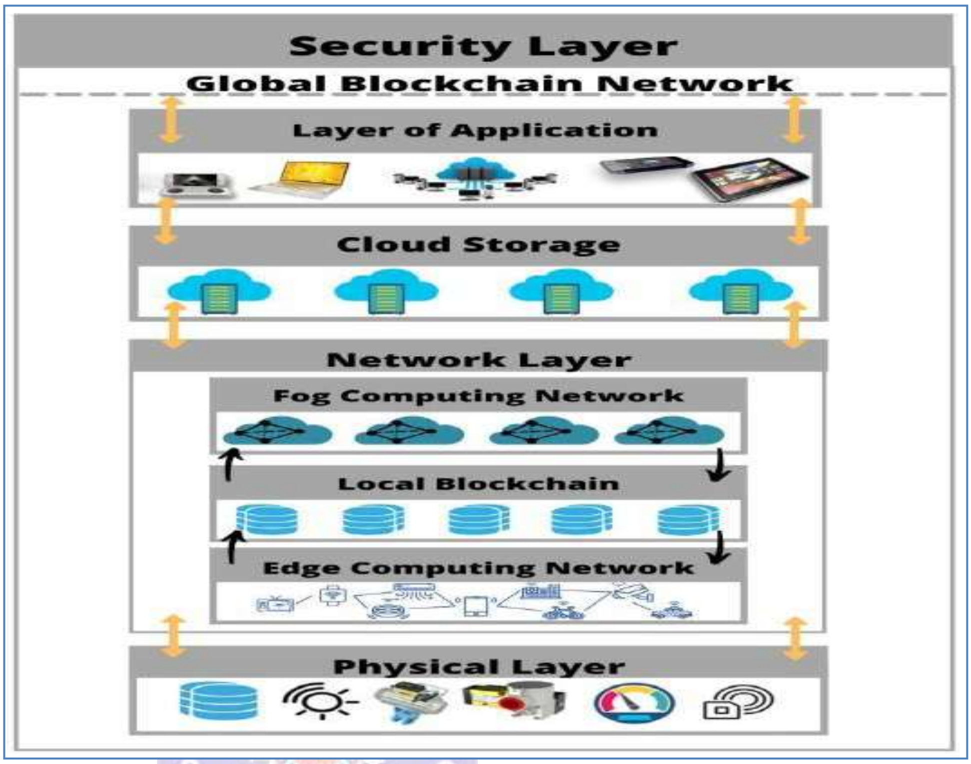

Fig. 1.4 BEFC-IoT Architecture

Internet of Things (IoT) sensors can record data generated by IoT-related applications, and they can be situated anywhere with an internet connection. Various types of IoT data can be directed to the right place for additional investigation based on the output criteria. Due to their proximity to the IoT components, fog or edge computing nodes must process such data as rapidly as possible. It is possible to direct less important data that does not depend on timely delivery to different nodes so that they can be processed and reviewed later.
To make Internet of Things (IoT) devices more secure, the blockchain layer is added. Blockchain technology allows for the immutable recording of all asset and transaction registries among numerous peers in a distributed, distributed, shared, and decentralised database directory. For trustless validation, the data is securely stored in chain blocks with timestamps mined. Distributed trust is a feature of the blockchain since it contains a complete history of transactions. Even though they are trustworthy now, trusted third parties and organised networks might still be abused, breached, or interrupted. They can also act unethically in the future. All transactions in the shared public ledger are actively tested and validated by a network of computers called mining nodes in a blockchain. Data stored in blocks should be permanently stored after authentication and agreement verification of transactions. It can be inferred from this that the data is immutable.
A local blockchain network and a global blockchain network should be separated at the blockchain layer to improve security and decrease latency. Due to the fact that not all of the micro IoT devices on the network will be able to manage the blockchain's load, the former will establish a centralised P2P communication between the IoT devices and a local blockchain linked with local storage within the private and trusted network. The second option, on the other hand, will set up a blockchain network that includes all of the fog/edge servers in the area, with the local BC serving as a node in this network. The network speed will be enhanced, and robust, secure communication will be formed between the fog/edge nodes and the IoT nodes, with low latency. Through the fog/edge network, the cloud is able to fulfil any supplementary demand for storage or other resources.

Fog Network: In order to reduce network overhead and latency, the nodes in a fog network are arranged into clusters, and each cluster is led by an appointed head. There should be no penalty for nodes due to excessive delays. Nodes are free to switch clusters if they experience excessive delays, and they may even choose a new leader for the cluster whenever they like. Every node in the cluster stores the public keys of the people making and receiving requests, as well as a forward list. Access transactions are recorded on the global blockchain, and all fog network cluster heads use multisig transactions relayed through cloud storage. Each cluster head decides for itself whether to keep or delete a new block based on its contact with participants of the obtained exchange. However, there are cases where it's more expensive to know where a certain block or transaction is located. For customers with many closed-edge networks who wish to manage them centrally, a pooled fog network can be formed by combining high-resource devices from different networks. In this mutual fog network, a standard miner and shared storage are selected. Every single transaction in the fog network starts at an edge system and goes to another fog network destination. Due to the consequences of double-spending, this causes a decentralised blockchain to fork, which is not permitted. Even with a fog network in place, the nodes in the component network that handle the most complicated network resources maintain a record of each block's number and its outcomes.

Cloud Storage: Sometimes, devices on closed-edge networks want to put their data in the cloud so that a third party may access it and give smart services. Users' data is organised by the cloud storage system into distinct chains, each with its own unique block number. Using the data's hash and block number, the buyer verifies their identity. If the desired data can be found inside the specified block number and the storage has the specified hash code, then it is considered authenticated. A hash function is used to guarantee the integrity of user data packets, which are sequentially sent and processed in memory. The generalised Diffie-Hellman method encrypts all blocks and finds a key for the new encryption by processing the files. This is due to the fact that no one else can learn the combination block number unless they possess the key. We can ensure that no one other than the genuine user can link the data and the new chain details to a traditional ledger due to the hash's collision-proof nature and the fact that only the authorised consumer knows the block number. Notably, customers have the option to create individual ledgers for their edge network devices or to combine all of the data into a single ledger. The customer will find this data useful if they wish to authorise access to any files on a particular computer. Fog nodes have limited processing and storage capacity, making it difficult to provide all end-user services at once. One possible solution to this issue is to provide each end-user with an allocation function that checks if the resource need has been fulfilled. Equation (1.1) can be used to express this.

$$(res) = \{ \begin{array}{l} \log(res) \,\forall\, \mathrm{res}(0, \mathrm{res}_{\mathrm{min}}) \\ \\ \log(res_{max}) \,\forall\, \mathrm{res}(\mathrm{res}_{\mathrm{min}}, \mathrm{res}_{\mathrm{max}}) \end{array} \qquad (1.1)$$

in which case, In this context, "A" stands for the allocation function, "res" for the total number of resources assigned to a job, and "resmin" and "resmax" for the minimum and maximum resources needed to fulfil service requests, respectively. Equation (1.2) shows the overall resource allocation by all edge network devices, which may be calculated using this allocation function.

$$A_{total} = A_k(res).P_k \qquad (1.2)$$

where A_{total} is the sum of all the resources that have been distributed to the devices or nodes at the edge. For the kth edge network node or device, the priority numbers and resource allocation functions are P_k and A_k(res), respectively. Individual nodes or computing setups at the fog or edge may only be able to supply the needed amounts of service when nearby computing setups have excess resources, highlighting the concept of scarcity and partial availability in many IoT scenarios. As the blockchain grows, nodes in close proximity to one another can record each other's storage and, depending on their connections to nodes in the network's periphery, send more data back to end users. In other words, nodes in close proximity to one another will assist one another by serving as edge nodes and sharing blockchain data. The fog and edge nodes can work together to help end users by sharing computing and storage resources. This is made possible through network connectivity. If the requested services are insufficiently provided by the accessible nodes, then... If that's the case, as shown in Eq. (1.3), the fog and edge nodes will distribute the processing of any background or non-critical services to their neighbouring nodes using their available resources.

$$res^x_{avail} = res^x - \sum_{k=1} res^k_{max} \qquad (1.3)$$

1.5 Evaluating BEFC-IoT Architecture Against Current Architecture Solutions

Table 1.1 shows the results of a comparison between the most popular architectures covered in this study and the proposed Internet of Things architecture, BEFC-IoT. Scalability, strong connectivity, big data, low latency, multi-level security, extensive network coverage, and network reconfiguration are some of the commonalities required by a wide range of Internet of Things (IoT) applications. All of the aforementioned characteristics cannot be met by the current design. Nevertheless, BEFC-IoT meets all the requirements, as shown in the research reviewed in this paper. Many Internet of Things (IoT) designs incorporate super technologies, either singly or in combination, such as cloud storage, edge computing, fog computing, and blockchain. To meet all the standards and requirements of the next generation of the Internet of Things, it is necessary to integrate these technologies into a single architectural solution.

Table 1.1 Comparison of BEFC-IoT Architecture with Existing IoT Architectures

	Scalable	Robust Connectivity	Huge Data Size	Low Latency	Multi-level Security	High Network Coverage	Network Reconfiguration
Three-Level Architecture	✕	✕	✕	✕	✕	✕	✕
QoS Based Architecture	✕	✔	✕	✕	✕	✕	✔
IoT Blockchain Architecture	✕	✔	✔	✕	✔	✕	✕
Service Oriented Architecture (SOA)	✕	✔	✕	✕	✕	✕	✔
Five Layer Architecture	✔	✔	✕	✕	✕	✔	✔
Cloud and Fog Based Architectures	✔	✔	✕	✔	✕	✔	✔
CloudThings Architecture	✔	✔	✕	✔	✕	✔	✔
Fog Computing IoT Architecture	✔	✔	✕	✔	✕	✔	✔
Edge Computing Architecture	✔	✔	✕	✔	✕	✔	✔
ICN-IoT Architecture	✔	✔	✕	✕	✕	✔	✔
SDN-Based Architecture	✕	✕	✕	✕	✕	✕	✔

IoT-A Architecture	✗	✓	✗	✗	✗	✓	✓
S-IoT Architecture	✗	✓	✗	✗	✗	✗	✗
BEFC-IoT Architecture	✓	✓	✓	✓	✓	✓	✓

Note: ✓ - Yes ✗ - No

1.6 Discussion

When it comes to security, modern IoT technologies are completely helpless. Reasons for this include insufficient funding for IoT devices, undeveloped requirements, and unstable hardware and software components, development, and deployment, as well as restricted capital in IoT devices. In this work, we offer a new layout that takes into account the data generated by current technologies and their possible demands. A new design has been suggested that makes use of state-of-the-art technological solutions, such as blockchain, cloud storage, fog computing, and edge computing. With its scalable and resilient design, this architecture can handle massive volumes of data, minimal latency, and extensive network coverage while maintaining many layers of security. More in-depth analysis of the enlarged BEFC-IoT Architecture's features, extra security measures, and the possibility of incorporating them into BEFC-IoT will be conducted in future research. The proposed design can be used as a foundation for creating models and frameworks for scalable IoT systems that use blockchain technology. These systems can handle large amounts of data, have reliable connections, and have minimal latency. In addition to assisting with network reconfiguration, this technology offers great network coverage and multi-layer security. Additional investigation into the suggested approach can be conducted using mobile IoT frameworks for both low-mobility and high-mobility models. As far as validating this architecture for framework development goes, IoV is the gold standard.

References

1. Yin, L., Hong, W. C., & Chao, H. C. (2018). Fog and edge computing: principles and paradigms. John Wiley & Sons.
2. Atzori, L., Iera, A., & Morabito, G. (2010). The internet of things: A survey. Computer Networks, 54(15), 2787-2805.
3. Verma, P., & Kaushal, S. (2020). A survey of fog and edge computing: Concepts, applications and issues. Internet of Things, 10, 100186.
4. Xu, Y., Sheng, Q. Z., Zhang, L., & Dustdar, S. (2018). Microservices architecture for internet of things systems. IEEE Internet Computing, 22(1), 12-21.
5. Patel, N., & Patel, D. (2020). Blockchain: An enabler for the integration of edge computing and IoT. Journal of King Saud University-Computer and Information Sciences.
6. Dorri, A., Kanhere, S. S., Jurdak, R., & Gauravaram, P. (2017). Blockchain for IoT security and privacy: The case study of a smart home. In 2017 IEEE international conference on pervasive computing and communications workshops (PerCom workshops) (pp. 618-623). IEEE.
7. Zyskind, G., Nathan, O., & Pentland, A. (2015). Decentralizing privacy: Using blockchain to protect personal data. In 2015 IEEE Security and Privacy Workshops (SPW) (pp. 180-184). IEEE.
8. Fan, K., Wang, W., Ren, Y., Liang, K., & Samarati, P. (2018). An overview of the state-of-the-art privacy-preserving approaches in the e-health clouds. IEEE Transactions on Cloud Computing, 8(4), 935-948.
9. Abeyratne, S. A., & Monfared, R. P. (2016). Internet of things (IoT) and edge computing for healthcare information systems. In 2016 18th International Conference on e-Health Networking, Applications and Services (Healthcom) (pp. 460-463). IEEE.
10. Dubey, H., Jain, A., & Varadharajan, V. (2020). Secure fog-based architecture for healthcare IoT using blockchain. IEEE Internet of Things Journal, 7(7), 6535-6542.