IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Edge Computing and Its Convergence with Blockchain in 5G and Beyond: Security, Challenges, and Opportunities

Showkat Ahmad Bhat[1], Ishfaq Bashir Sofi, and Chong-Yung Chi[1], (FELLOW, IEEE)
[1]Institute of Communications Engineering, Department of Electrical Engineering, National Tsing Hua University, Hsinchu 30013, Taiwan

Corresponding author: Showkat Ahmad Bhat (showkatbhat1994@gmail.com)

This work was supported by the Ministry of Science and Technology (MOST), Taiwan, under Grants MOST 108-2221-E-007-012 and MOST 109-2221-E007-088.

ABSTRACT The internet is progressing towards a new technology archetype grounded on smart systems, heavily relying on artificial intelligence (AI), machine learning (ML), blockchain platforms, edge computing, and the internet of things (IoT). The merging of IoT, edge computing, and blockchain will be the most important factor of empowering new automatic service and commercial models with various desirable properties, such as self-verifying, self-executing, immutability, data reliability, and confidentiality provided by the advancement in blockchain smart contracts and containers. Motivated by the potential paradigm shift and the security features brought by blockchain from the traditional centralized model to a more robust and resilient decentralized model, this tutorial paper presented an multi-tier integrated blockchain and edge computing architecture for 5G and beyond for solving some security issues faced by resource-constrained edge devices. We begin with a comprehensive overview of different edge computing paradigms and their research challenges. Next, we present the classification of security threats and current defense mechanisms. Then, we present an overview of blockchain and its potential solutions to the main security issues in edge computing. Furthermore, we present the classification of facilitating developers of different architectures to select an appropriate platform for particular applications and offer insights for potential research directions. Finally, we provide key convergence features of the blockchain and edge computing, followed by some conclusions.

INDEX TERMS Cloud computing, blockchain, edge computing, mist computing, fog computing, security and privacy, 5G, cloudlets, server-less computing, consensus process, smart contracts, and blockchain platforms.

I. INTRODUCTION

With the precipitous advancement of the IoT, billions of devices are being linked with the network. The number of devices connecting to the system is an amazing thing but even more than that it is not the devices but the amount of data that is just growing exponentially. As the devices are coordinating with each and everything around in the network, data traffic is continuing to grow enormously. It is expected about 2 GB of traffic per day per person by 2022, which is a lot but that is going to be completely dwarfed by the data generated by devices. Some examples of that are such as a connected airplane that generates 5 terabytes per day, a connected hospital 3 terabytes per day, a smart factory 3 petabytes per day, and an autonomous vehicle that generates 4 terabytes per day. With all such data being generated we cannot simply transport them into the cloud to analyze. The fact is that no matter how fast our uplink is, we cannot transport such a quantity of data with the available bandwidth. All these factors act as motivation for academicians and industries to work towards next-generation cloud computing technologies. In addition to this, almost all these edge devices are effortlessly hacked and compromised. Usually, these edge devices are constrained in computational complexity, data storage, and the network resources, and thus are highly susceptible to security attacks than other edge nodes such as smartphones, computers, or tablets.

Cloud computing over the years has developed an essential part of data processing. Though, the cloud servers deployed centrally on a global scale have to compute a

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

massive volume of data. Moreover, the latency of the network also surges with an increase in the physical distance between the end-user and the cloud, thereby resulting in the rise of response time and quality of service (QoS) degradation of user applications. Furthermore, the user device performance has a significant impact on the computing time in this environment. As we move to next-generation computing technologies, we are going to witness sorts of storage, computation, and analytic capability extend out to the edge devices.

A. OVERVIEW

Edge computing paradigm performs by letting some of the data computation to be executed by an edge server located close to the data generator such as mobile devices, sensors, smart homes, etc., increases the performance of low latency real-time applications and thus reduces the workload of cloud servers. Autonomous cars are one of the great examples of real-time edge computing applications [1]. It is going to look like a mini cloud or data center on wheels architecturally but on a different scale. It is going to have all the capabilities to store, computing, and analytics locally along with performing higher computational tasks and functions at the cloud. The autonomous car has gotten cameras, radars, LiDAR, GPS, etc. along with storing all the produced data. The car needs to take real-time decisions on how to steer the car. It simply cannot direct the data to the cloud and wait for the response to which way to steer the car, which is not going to work. The latency requirement is way too short to ever do that. We can also consider the cases of a connected factory or a hospital. Due to privacy concerns, the factory or hospital does not intend to send all the data to the cloud but needs to do some analytics locally for instance removing or masking the privacy critical data before sending it to the cloud.

So evolving to such architecture where we have acquired storage, computing, and analytics distributed across the end-to-end network and drive some interesting new capabilities and technologies. Speed, security and privacy, scalability, location of computing resources, and low cost are some of the driving forces to push some of the functionalities to be performed on the edges.

The existing supporting technologies that are going to be used within the edge computing network architecture are listed in Figure 1. Containerization, Orchestration, 5G, AI & Machine Learning, etc. are the technologies that are being used currently, and in these scenarios, it is just adaptation of these technologies in edge computing framework [2]. Accordingly, there is a vast set of open-source frameworks of these technologies. But different use-cases of edge computing have different scale requirements. Hence, adaptation and optimization need to be done in these supporting technologies to integrate with edge computing. Edge computing is going to become an indispensable component of the 5G network. If we think about autonomous vehicles and how they communicate with the network and one another, 5G is going to have a critical part. Software-defined network (SDN) space has a bunch of essential open-source technologies as we can never build a 5G network employing existing expensive and slow hardware and software. These technologies are having significant roles to play there.

Since the edge computing paradigm exemplifies a collection of interconnected networks and heterogeneous devices. It inherits the conventional security and privacy issues related to all the constituent technologies. And these threats are, in fact, very substantial. Along with securing all these constituent components, we also need to orchestrate the assorted security techniques. After the cloud-like computing and analytics functions are brought to the network edge, novel security situations will arise which are yet to be extensively studied [3]. These security issues are considered a blend of the "worst-of-all-worlds" related to security.

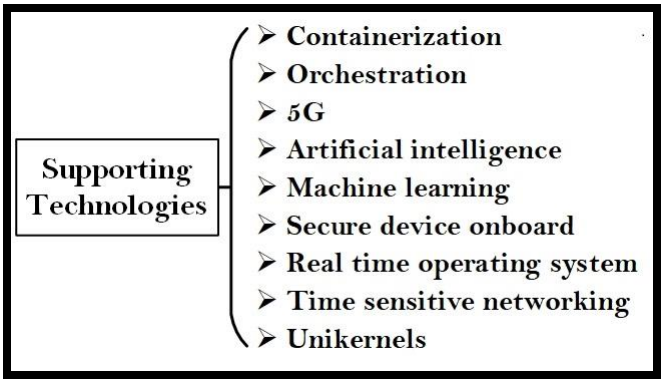

FIGURE 1. Supporting technologies of edge computing

The memory, power, and computing resource constraint devices pose additional challenges to the edge security. Thus the security solutions need to be adapted in the constrained edge architecture. Blockchain is being considered as a disruptive technology by academicians and industries that offers potential solutions in managing, controlling, and most importantly in the security of edge computing networks and devices [4]. The incorporation of blockchain and edge computing into a single framework will make it possible to have reliable access and control over the network, storage, and distributed computational resources at the edge. Consequently, edge computing makes storage, network management, and computation capabilities available close to the edges in a safe manner. Regardless of the prospect of an integrated blockchain and edge computing network, its scalability augmentation, self-organization, the convergence of resources, resource management, and the novel security challenges [5] remain open before its application in edge use cases.

B. RELATED WORK

There exist a lot of discussions in the literature of edge computing and blockchain technology but very little work has been done towards the integration of edge computing and blockchain technology at the edge nodes. For edge

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

computing, various studies on security of several edge computing archetypes have been reported, e.g. fog computing [6-10], mobile edge computing [10-12], mobile cloud computing [13-15], and mobile ad-hoc cloud computing [16-18]. These preliminary analyses examine security threats that influence the integrity of these edge computing archetypes, together with an overview of the security procedures of defending all functions and infrastructures. Other surveys have studied basic characteristics, research challenges, and opportunities of different edge computing paradigms [19-22]. Caprolu et al. [2] discussed security issues about supporting technologies for the edge computing, and Dustdar et al. [23] presented detailed individual characteristics and use cases along with certain future challenges of edge and fog computing paradigms. Alrowaily et al. [24] reviewed the concepts, characteristics, security, and edge computing IoT-driven applications besides the security features of a data-driven world. Wei Yu et al. [25] conducted a broad review and examined the edge computing to enhance the implementation of IoT, and classified the edge computing in different groups based on their architecture. In addition to security issues in edge computing, they also evaluated the accessibility, reliability, and the confidentiality of each group, and proposed a framework to evaluate the security of the IoT networks involving edge computing. Roman et al. [3] analyzed the security threats, research challenges, and mechanisms inherited by all edge computing archetypes, meanwhile pointing out the potential interactions and functions to integrate security mechanisms in edge paradigms. Yahuza et al. [26] investigated existing works and highlighted different categories of privacy and security threats and state-of-the-art technologies for the curtailment of different security threats. They also summarized different metrics to evaluate the performance of these techniques, classified different attacks, and presented some technical trends in alleviating these attacks.

Recently, blockchain technology has received extensive attention, and numerous studies have been reported. Some works [27-31] presented the main principles of blockchain technology and its applications especially bitcoin. Other works can be categorized into three types: applications [32-40], security threats and privacy issues [41-46], and consensus protocols [41], [47-50]. Eyal et al. [27] proposed a novel Bitcoin-NG (next generation) designed to scale blockchain protocol and meanwhile it is byzantine faults tolerant and robust against extreme churn. Pilkington et al. [28] studied the main principles of blockchain technology with some more applications. Drescher et al. [29] presented comprehensive conceptual and technical aspects of the blockchain necessary to build a blockchain and understand business-related blockchain applications. Besides, [29] also analyzed the economic impact and potential of blockchain technology in a wide range of applications. Cachin et al. [30] presented a hyperledger blockchain fabric architecture, which is an open-source permissioned distributed ledger framework based on user-defined smart contracts. It has robust security, identity features, and employs a segmental architecture along with attachable consensus protocols. Maesa et al. [31] reviewed the certain application of blockchain technology in particular healthcare ledger management, identity management systems, access control system, electronic voting system, and distributed notary management. For each application, [31] also evaluated the issues, associated requirements, and potential advantages that blockchain technology implementation might bring forth.

Recent related works in developing distributed platforms for the IoT emphasized the tendency of optimizing the blockchains for their implementation onto the resource-constrained IoT edges. Ali et al. [51] specified the latest works in which IoT networks are isolated by using blockchain-connected gateways. Ouaddah et al. [52] presented a completely distributed pseudonymous and privacy conserving authorization controlling framework (called FairAccess) that empowers consumers to own and manage their data. Bahga et al. [53] introduced a cloud-blockchain hybrid system enabling industrial IoT (IIoT) devices to interact with both cloud and blockchain. This system needs more computational resourceful IIoT devices, equipped with single board computers (SBC) capable of providing an interface between these platforms. Khan et al. [54] presented major IoT security issues with regard to its layered architecture along with possible blockchain solutions and open research gaps.

Apart from isolated industry verticals, to prevent the scalability challenges faced during the blockchain and IoT integration, the use of multiple blockchains is a promising method. Recent developments in research put significant efforts on designing inter-blockchain networks for reducing the largely incoming transactions to any particular blockchain. Sagirlar et al. [55] described hybrid-IoT layered blockchain architecture, for which Proof-of-work (PoW) and blockchain inter-connector systems are amalgamated in the hybrid-IoT framework to enable communication between blockchains. However, it cannot be implemented on resource-constrained IoT edge devices because of the high computational necessities of PoW consensus. Xiong et al. [4] presented the concept of edge computing for the applications of mobile blockchain, particularly for IoT blockchain mining task offloading on a testbed. Stanciu et al. [56] examined the IEC 61499 standard for decentralized control systems and reported the ongoing research related to the implementation of functional blocks as smart contracts on a management level by using the blockchain technology platform. The convergence of edge nodes that carry out the responsibility of process control at the executive level is built on the micro-services framework. Conoscenti et al. [57] and Ali et al. [58] deliberated the matters of scalability, reliability, and security and privacy in blockchain for the IoT applications. Dorri et al. [59] suggested a new blockchain multi-layer architecture for IoT using a two-layered superimposed public blockchain architecture and several smart home blockchains, and those

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

smart homes act as centralized private ledgers. The authors of [59] have developed a quick and scalable consensus mechanism by selecting random validators based on their reputation to decrease the computational overhead for the overlay blockchain. Mendki et al. [60] described several possible challenges in a blockchain-enabled edge that may arise for ensuring data privacy and data integrity meanwhile executing the data processing remotely. The authors of [60] also presented current research works and possible solutions to solve the privacy and integrity challenges. Bhattacharya et al. [61] reviewed the mobile edge computing (MEC) architecture and proposed a mobile blockchain framework to ease the mining process. They also investigated the effects of the convergence of blockchain with MEC. Researchers and industries also concentrated on direct acyclic graphs (TDAG) based approaches to blockchains such as IOTA [34] and NEO [62]. IOTA tangle has a great potential in enabling a decentralized IoT edge by means of a distributed ledger framework. Efforts are going on to solve and reduce its existing limitations. One of these endeavors is G-IOTA [63], which expects to enhance the tip selection algorithm of the IOTA tangle for the new transaction issuance and validation.

C. CONTRIBUTION

Previous methodical surveys and system reviews have identified earlier developments, though development in the field of edge computing needs to reconsider the paradigms (Blockchain, AI, ML, and IoT) which are going to drive the edge computing. There is a necessity for methodical survey for evaluation, upgrade, and integration of current research in the edge computing field with the emerging technologies and archetypes e.g. blockchain, IoT, sever-less computing, and AI. The main contributions of this tutorial work are:

- A comprehensive overview of different research challenges of edge computing along with security issues and their current defense mechanisms.
- Identification of different challenges and risks of blockchain technology.
- Identification of the diverse means to integrate edge computing and blockchains along with the impact on edge computing and related archetypes.
- Some blockchain-based solutions for numerous security-related challenges in edge computing.
- A conceptual integrated blockchain and edge computing architecture for 5G and beyond, which is robust against different security attacks together with the influence of blockchain on edge computing evolution.

The remainder of the paper is structured as follows. Section II outlines different edge computing paradigms and the proposed architecture is detailed in the case study subsection. Section III discusses challenges, emerging trends, and impact areas of edge computing. Section IV outlines the classification of security threats. Section VII reports blockchain as a potential solution for different security challenges. Figure 2 shows the complete organization of the tutorial paper.

II. EDGE COMPUTING

Edge computing is a process of deploying data computing capabilities near the edge of the network, where data is being generated and actions are performed to enhance response time and bandwidth utilization. Figure 3 shows the basic three-layer (cloud data centers, fog computing node, and edge device layers) infrastructure of edge computing.

A. EDGE COMPUTING CORRELATED MODELS

Cloud computing technology is not capable of meeting certain requirements due to several reasons for instance jitter, low latency, mobility support, energy, and context awareness that are essential for many applications and services, for example, augmented reality, health services, autonomous vehicular networks, etc. In recent years, different paradigms have been developed to fill these requirements such as fog computing, mobile cloud computing (MCC), mobile edge computing (MEC), and server-less edge computing. This section discusses the fog and various edge computing paradigms. Some key features that distinguish cloud, fog, and edge computing are mentioned in Table I.

1) FOG COMPUTING

The fog computing model was designed to create a decentralized computing architecture and extend the cloud-like applications, services, networking, storage, computing capabilities, redundant data removal, data offloading, and decision making on the edges. Thus preserves time and communication resources of the network [3], [6], [64]. Hierarchical infrastructure is created by taking advantage of fog computing architecture, where the locally generated data is computed at the fog node and the global analytics are executed at the cloud servers [65]. It is needed to decrease the overheads of the transmitted data, and subsequently, enhance the performance of the system by decreasing the required processing and storage of huge amounts of superfluous data in cloud platforms. For example, GPS data compression can take place at the edge device before offloaded to cloud servers in an intelligent transportation system (ITS) [66]. Fog computing is also well-defined as horizontal and vertical platforms [67].

Fog platform can be combined with wireless mobile communication technologies beyond 5G as random access network (RAN) which is defined as Fog-RAN [68]. To facilitate faster content retrieval and reduce the workload on the front-haul, F-RAN computing resources can be utilized for accumulating at the network edge.

2) MOBILE EDGE COMPUTING

The objectives of mobile edge computing (MEC) is to reduce system latency and improve the performance of the network by installing mobile applications at the edge and optimize the current mobile network architecture. IBM and Nokia were the first to create this application platform at the mobile base station and provide the services from the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

edges of the network [3], [69]. Under the specifications provided by the industry specification group (ISG) launched by the European telecommunications standards institute (ETSI), it aims to offer IT service atmosphere and cloud-like computing abilities at mobile edges [70], [71]. A heterogeneous MEC platform creation is being pursued by the ISG, where several services providers can deploy their applications and services. These service environments will be deployed by telecommunication companies in their existing infrastructure once the standards are established to offer low latency, high bandwidth efficiency, quality-of-service (QoS), routing area code (RAC) information, and location awareness [72], [73]. Augmented reality, unmanned aerial vehicle (UAV) based mobile computing system [22], connected cars, intelligence video acceleration, and IoT gateways are some of the expected application of the MEC [74]. Sharing of resources amongst several mobile service providers such as infrastructure-as-a-service (IaaS) [19], as well as storage and computing abilities. For enhancing management flexibilities of MEC can be done by virtualization [75-77] of both mobile edge and mobile edge network.

MEC helps in enhancing current applications and presents tremendous potential for the development of novel wide-ranging services and applications. The main use cases of MEC are computational offloading [78], distributed content delivery, caching [79], web performance enhancement [80], application-awareness, and content optimization, etc.

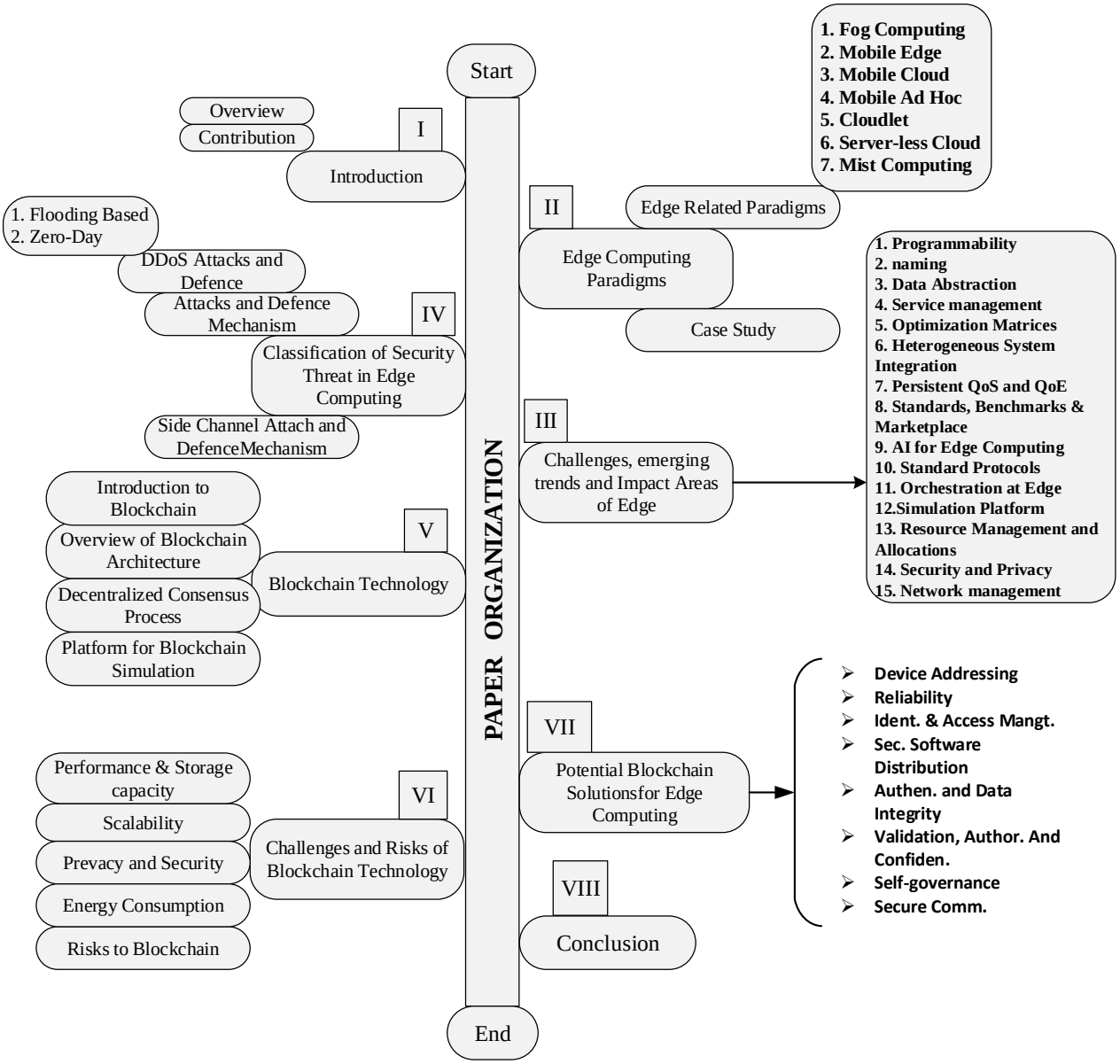

FIGURE 2. Structure of the paper: Edge computing and its convergence with blockchain for 5G and beyond

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

3) MOBILE CLOUD COMPUTING

The development of the mobile cloud computing (MCC) paradigm has changed the existing technology widely from physical to virtualized network infrastructure [81-85]. MCC is demarcated as a platform on which data processing and storage are implemented in the mobile clouds, instead of handling on smart mobile edge devices [86]. Users subscribe to the services from the mobile cloud by offloading the high computational intensive applications to mobile clouds for processing and the required data or information is extracted and send back to the smart mobile device (SMD). The subscriber needs to pay for the services they subscribe to, from the service provider [87]. The pay-as-you-go concept of charging attracts smartphone users and subscribes to services namely IaaS, system-as-service (SaaS), and platform-as-a-service (PaaS).

Mobile cloud computing services are independent of the location and mobility of the mobile device, but the user can connect MCC through a web browser at any time from anywhere without any interruption [88]. MCC has reduced several limitations for instance data processing and storage faced by mobile communication architecture. Both processes are executed using cloud resources, by connecting with the cloud over the internet utilizing Wi-Fi, LTE, 5G, or any other wireless technology through there SMD. Accordingly, MCC can be defined as a bridge between cloud computing and users via the mobile web. Therefore, MCC is the convergence of three technologies smart mobile edge devices, mobile communication networks, and the cloud computing platform.

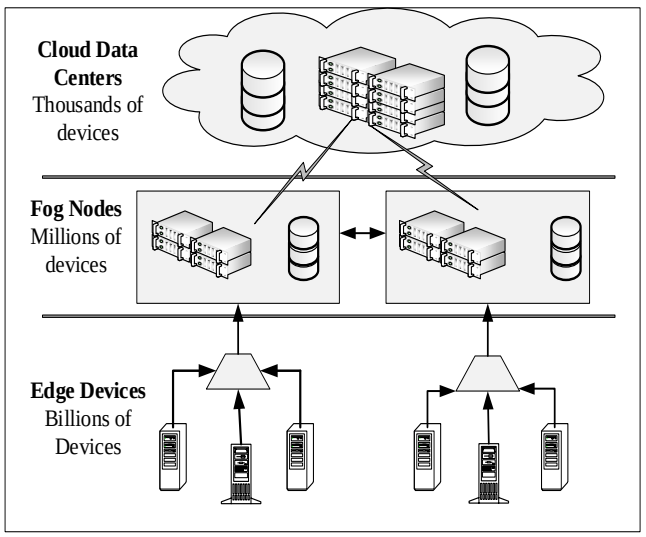

FIGURE 3. The edge computing infrastructure

4) MOBILE AD HOC CLOUD COMPUTING (MACC)

Workload offloading in the MCC paradigm solves many key problems such as resource constraints, computing capability, and energy limitations, but it has been found that connection to the remote cloud is not always accessible and gives rise to the development of MACC [18], [89], [90]. It utilizes the resources of available devices to develop a dynamic ad hoc distributed system topology to execute a common task. Thus, a MACC can be described as a distributed network of autonomous computers, smartphones, and other IoT devices connected via a communication network and distributed middleware, allowing devices to synchronize their functions and share resources within the network [90], [91]. Due to the highly dynamic nature, the system must make provisions to accommodate the users that constantly leave and join the system.

MACC is a virtual supercomputing edge [92] having some distinctive features e.g. dynamic topologies, finite resources, variable link capacity, restricted physical security, and power-constraint operations mainly inherited from the mobile networks and making it exceptional from other server-based cloudlets and remote clouds. Since, due to the battery-operated nature of the devices, only power constraint computations are executed in the MACC. Distributed and centralized are the two execution models for the MACC. The design of lightweight algorithms and frameworks are required for mobile ad hoc networks.

5) CLOUDLET COMPUTING

Cloudlet is typically a viable personal computer or a workstation [7] having a virtual machine running on it [93-95]. The purpose of the cloudlet is to decrease the end-to-end latency between the cloud and mobile devices.

In the cloudlet computing architecture, all the heterogeneous devices present nearby including mobile phones, laptops, and fixed computers cooperate and form a cloudlet [96]. Thus cloudlet is also a heterogeneous network. Dynamic architecture is scalable, allows devices to leave and join the network on the go. Cloudlets are also referred to as microdata centers introduced first by Microsoft as a replica of traditional cloud computing data centers for edge devices.

There are several successful functional examples of commercial cloudlet services in practical environments [97]. Cloudlets are evolving to be an important enhancement to the MCC architecture. It brings the users closer to the cloud. Resource capabilities of the mobile devices are augmented effectively by data offloading and computation offloading [98]. On the other hand, there are several research challenges for the cloudlets to be deployed widely. Security of the cloudlets is the prime importance for the users to subscribe to privacy-related services, e.g. e-Commerce and e-banking. To resolve the security-related issues, a reliable security mechanism needs to be developed. Furthermore, there are no standards for the cloudlet services and to manage the large number of services and operations the “killer app” for cloudlet mobile computing is yet to be developed.

6) SERVERLESS CLOUD COMPUTING

Function-as-a-service (FaaS) or serverless computing is well-defined as software architecture to separate services and applications into several functions and provide a smooth holding and execution environment as a platform.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

The software application designers are only concerned about the lightweight and stateless functions having the capability to run over application program interface (API) based on the on-demand principle, consuming resources only at the point of function execution [99], [100]. In applications based on serverless computing business logic is located at the end and is made available to the user through mobile or web applications to execute on provided resources with no need of renting virtual machines (VM) while the data storage, application logic, and servers are situated in the cloud [101]. FaaS can overcome or address several open research challenges e.g. fault tolerance, load balancing, and resource allocation.

Moreover, two types of services (FaaS and backend-as-a-service (BaaS)) are provided by the serverless computing, and Google Cloud, Amazon AWS, and Microsoft Azure support these services. The serverless computing paradigm comes up with many research issues such as bandwidth consumption, task scheduling, security and privacy, server tools, declarative deployment, refactoring functions, concurrency, and recovery semantics, and code granularity. Some future research directions for FaaS are precise as the development of new IoT based applications for supplementary secure communication and enhance the privacy of data, due to resource constraint devices and inability to support heavy security algorithms and firewalls. So blockchain technology needs to be implemented to improve security and the AI system is the other potential technology to enhance the serverless computing design.

7) MIST COMPUTING

By collaborating fog and cloud computing based on the notion that communication between sensors and actuators level should be made possible without straining the communication networks and internet has given rise to the development of mist computing, by exploiting the resources of the network from the devices at the very edge [102]. Thus mist computing is explicitly defined as a model in which devices having anticipated accessibility, distribute computational and communication resources as services in their surroundings via the device-to-device (D2D) communication protocols [103]. Mist nodes are different from the regular mobile web servers (MWS) offering static software services to subscribers. Some core features of mist computing are that it is scalable, reconfigurable, self-location aware, and uses machine-to-machine (M2M) communication [104]. In this paradigm, any node can subscribe to the services offered by any other device by just having a working internet connection [105]. From the above inference, mist computing is evolving rapidly and is considered a potential technology to solve several challenges e.g. a single-point failure in cloud and fog paradigms.

B. CASE STUDY

Real-time performance guarantees are critical for new technologies that rely on real-time virtualization including applications such as safety features in smart cars [1] designed to prevent accidents or remote monitoring of patients [106] with serious conditions or others. Edge computing combines the characteristics of cloud computing and virtualization to bring high computing competencies as close as conceivable to end-users. The commercial potential of this technology is massive; however, the practical implementation of this technology is dependent on its maturity as well as its definite description by appropriate standardization bodies, commercial organizations, and open-source projects. Thus, we presented a multi-tier blockchain-enabled edge computing architecture [4] shown in Figure 4.

The architecture in Figure 4 represents the applicability and efficiency of blockchain in mobile and static edge computing environments. All the devices in this network architecture gather, store, provide services, and communicate data by running blockchain-enabled applications through transactions. The blockchain mining process will be used to verify all the operations.

TABLE I. KEY FEATURES THAT CONTRAST EDGE COMPUTING, FOGE COMPUTING, AND CLOUD COMPUTING

Key features	Cloud computing	Fog computing	Edge computing
Architecture and geo-distribution	Centralized computing model	Distributed computing model	Distributed model (Fog subset)
Security	Less security	Improved security	High security
Purposes	In-depth data analysis	Quick data analysis	Real-time data analysis
Location of data processing	Central cloud server	Network gateway	On-device
Mobility	Limited support	Supports mobility	Highly supported
Response time	Slow	Improved response time	Fast response time
Scalability	Low	Higher than cloud	Higher than Fog
Analysis	Long-term	Short-term	Short-term
Location awareness	Not possible	Possible	Possible
Data	Industrial big data	Local network data, Microdata centers	Real-time data, Process-specific data
Storage	Unlimited	Limited	Constrained
Number of servers	Few	Large	Very large
Usage of bandwidth	High	Lower than Cloud	Low
Jitter	High	Low	Very low
Scope of network	Complete network scope	Limited to domain	Limited to application
Latency	High	Low	Very low
Computation capabilities	Higher	Lower	Very low

To solve the problem of increasing ledger size beyond memory capacity due to the high block generation rate, the edge computing paradigm is used as ledger storage. Thus,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

transaction verification and block generation will be performed in the traditional way of blockchain, though, the verified transactions, the performance records, the node details, and the communication between nodes will be stored at the edge computing nodes present inside the network. Thus, the IoT devices will function appropriately as blockchain nodes, though, these nodes will export and store ledger to edge computing node at every stage of each transaction. In the proposed architecture smart contracts are used for edge device registration, data storage, service, and resource management along with validating the transaction data within the network. Edge devices will be capable of accessing the ledger to keep it updated with each generated block. In such a way, access to the storage is faster, with low latency. Therefore, the convergence of blockchain with edge computing will increase the performance and security of IoT devices in particular, and the whole edge network in general.

The data abstraction procedure will be performed by edge devices according to the service needs of a particular application. Thus data related to devices like power status, accessibility, and physical conditions can be utilized to achieve the required QoS all the time. Besides, data associated with areas such as energy, healthcare, autonomous vehicles, factories, etc. can be mined from the stored data on edges as well, so as to help them to enhance the performance and reduce resource consumption.

III. CHALLENGES, TRENDS, AND IMPACT AREAS OF EDGE COMPUTING

We have characterized seven different edge computing paradigms and a case study in the preceding section. Even though the idea of edge computing is straightforward but its implementation poses numerous challenges. The main ones are described in detail and some challenges with their research directions are summarized in Table II in this section.

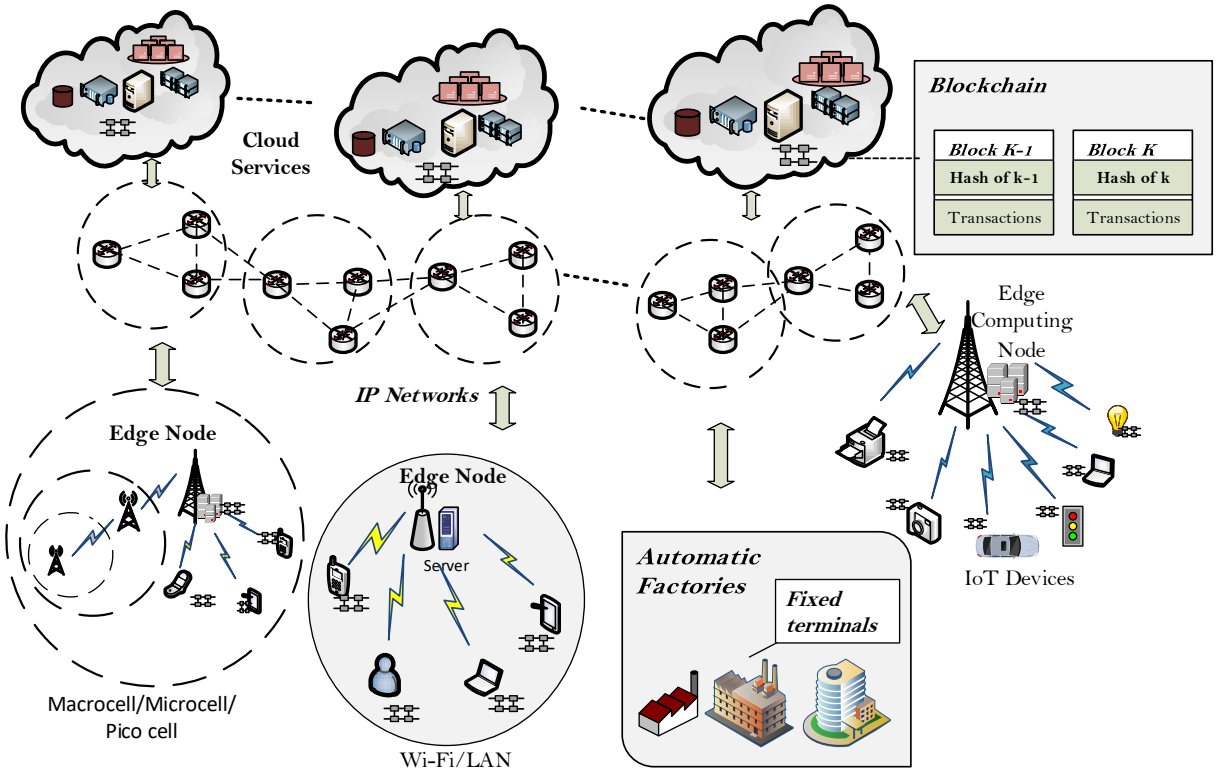

FIGURE 4. Proposed integrated blockchain and edge computing network architecture for 5G and beyond

1) PROGRAMMABILITY

Typically, cloud computing is capable of supporting large and complex codes due to the availability of appropriate infrastructure with high computational capabilities, storage, and power. In contrast, in the edge computing paradigm, the computation is done at the resource constraint end-nodes, which are most probably working on heterogeneous platforms. In this scenario, the programmer confronts setbacks to develop an application capable of running in the resource constraint and heterogeneous edge computing environment.

- *Frameworks and Languages*

With the recent development of the edge computing paradigm, general-purpose computing might not be supported by the edge nodes, the development of frameworks and toolkits will be needed. The software design that targets to run on certain edge nodes will require sustaining parallelism in workload and data while executing the workload on multi-layered hierarchical hardware of the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

network. The programming language used in the software design needs to respect the diversity of the hardware and resource capability of the system. Vendor-specific edge nodes make it more complicated than current models by taking into account the workflow supporting frameworks of each vendor that makes the cloud accessible. In a distributed environment, software frameworks, and programming toolkit development [107], [108] is a definite research direction.

- *Lightweight Libraries and Algorithms*

Due to hardware limitations, edge nodes are not capable of sustaining substantial and complex software applications and operating systems. For example, Intel's T3K concurrent dual-mode system on chip (SoC) with four core Arm-based CPU and very narrow memory on small cell BTS (base station), will not support to run heavy and complex data processing tool Apache Spark that needs at least 8 GB memory and 8 cores CPU to deliver a good performance. Edge node analytics will need lightweight programs to perform realistically machine learning and data processing workloads [109], [110]. There are merely some lightweight frameworks for instance Apache Quarks and Tensorflow; Apache Quarks can be deployed on some edge computing devices like smartphones to perform real-time data computation along with filtering and windowed combinations, not adequate for cutting-edge computing functions.

- *Micro OS and Virtualization*

Challenges associated with the setting out of applications and services on the heterogeneous edge nodes can be tackled by doing significant research toward micro OS and micro Kernels. Quick deployment, scale-down boot-up time, and isolation of resources are the anticipated benefits [111]. In literature, research suggests that mobile containers that combine with hardware through several virtual devices can deliver comparable functioning to built-in hardware [112]. Docker's like container [113], [114] technologies are at the final phase of development and allow the swift deployment of services [115] and applications on heterogeneous platforms. Significant research is needed to approve containers as an appropriate scheme for deploying applications on edge nodes.

2) NAMING OF EDGE DEVICES

There has been little work done in literature on the development and standardization of a powerful naming scheme for the edge (IoT) computing environments. To communicate with and among the heterogeneous devices practitioners must learn different communication and network protocols. The edge computing naming scheme will then handle the high changing network topology, security, and device mobility, along with the scalability of tremendously large unreliable devices. Traditional naming schemes, such as domain name system (DNS) [116], uniform resource identifier (URI), electronic product code (EPC), MobilityFirst [117], uniform resource locator (URL) and other uniform resource identifiers fulfill the maximum of the current networks very well. Named data networking (NDN) [118] is user-friendly and scalable for the service management and offers a hierarchically organized name for the data-driven network. Though extra proxy is required to concur into existing communication technology protocols such as Wi-Fi, ZigBee, or Bluetooth, and NDN has a security issue (difficult to separate device physical information from the network address). MobilityFirst is very proficient for highly mobile environments and can separate device names from the network address but it needs a global unique identifier (GUID) to be used for the naming, which is not required for static data aggregation applications such as smart home environments. It is also very difficult to manage services in MobileFirst for the edge as GUID is not user-friendly.

However, for edge computing naming schemes, it needs to be flexible enough to work for the heterogeneous edge network with highly mobile and resource-scarce devices. Though, sometimes internet protocol (IP) naming could be heavy for certain resource constraint edge devices due to its complexity and overhead. Therefore, for highly dynamic environment applications e.g. smart city level system, naming is an open research problem and needs further research by the academicians and researchers.

3) DATA ABSTRACTION

Due to a tremendous volume of raw information produced by billions of devices interconnected in the edge computing paradigm, it is becoming challenging to store and communicate all this raw data in the edge computing paradigm [119], [120]. Considering the IoT based smart-home context, data from the smart things must be computed at the gateway stage, to decrease the transmission cost, enhance privacy protection, remove the unwanted raw data, and increase event detection, and so on. Treated data will be communicated to the higher levels for future applications. The data abstraction process faces several challenges [121].

First of all, different devices produce data of different formats as shown in Figure 5. Due to the secrecy and privacy of the end-users, raw data must be made inaccessible to the applications running on the edge operating system (edgeOS). Besides, they have to pluck out the information they are concerned about from an incorporated table containing the processed data. But hiding the specific details of the sensed data reduces the usability of the data. Yet, we cannot store huge quantities of data due to storage challenges. Besides the reliability of the data reported by some edge nodes, due to insecure wireless communication channels, low precision sensor, and vulnerable environment. IoT application and system developers still face big challenges to abstract data from unreliable data sources. The naming of data from massive IoT devices is becoming a huge challenge. A huge volume of data is being generated and uploading by billions of IoT devices simultaneously, which is a challenging task for file nomenclature, file reliability management, resource allocation, etc. for different storage servers with diverse operating systems.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

TABLE II. CHALLENGES AND POTENTIAL RESEARCH DIRECTIONS OF EDGE COMPUTING

Functions	Challenges	Research directions
Scalability and elasticity	Hardware is heterogeneous. Resources virtualization, isolation, and performance Scalable Software. Middleware to monitor performance.	Virtualizing GPUs and novel architectures. Programming language and code abstraction. Approx. computing, performance, and cost trade-offs. Scalable distributed algorithms.
Resource management and scheduling	Auto-scaling and resource control. Multi-cloud operation and load-balancing. Ad-hoc design of multiple control loops. Sensitivity to errors in workload characterization.	Data analytics for resource management. AI-driven management. Function-level QoS management in serverless computing. Holistic management of data center and edge.
Reliability	Failure correlation leading to large-scale service disruptions. Lack of holistic service reliability models. Lack of automatic reliability aware service management mechanisms. Lack of failure-aware provisioning policies.	Failure is aware of resource provisioning. Reliability as a service. Efficient storage reliability. Reliability and energy efficiency correlation.
Sustainability	Energy needs of cloud data centers (CDC). Virtual machine consolidation to minimize energy consumption. Optimized scheduling of traffic flows between services. ML-based methods for task allocation. Energy versus QoS trade-offs.	Dynamic task scheduling for energy and QoS optimization. Building algorithms and architecture for distributing computing efficiently. The interplay between IoT-enabled cooling systems and the CDC manager. Renewable energy for the CDC.
Heterogeneity	Heterogeneity at VM, vendor, and hardware architecture levels. VM placement, provisioning, and scheduling. Hardware acceleration adoption with vendor-specific languages.	Predicting performance at the hardware architecture level. Management strategies that work across VM, vendor, and hardware architecture levels. High-level coding languages for abstraction and elasticity. Disaggregated datacenters.
Interconnecting clouds	Cloud interoperability. Common rules and standards for security and service quality. Cross-site virtual networking. Interoperation beyond minimum common denominator of services.	Data and application formats. Application customization. SDN and network functions virtualization (NFV) enhanced intercloud operations. Equivalent service compositions across providers.
Empowering resource-constrained devices	Mobile computing (MC) binding models-task delegation and code offloading. MC adaptability issues. Cloud-centric IoT. Cloud computing at the edge with fog.	Multi-tenancy in MC. Containers in edge computing. QoS is aware of application deployment. Edge analytics for real-time stream data processing.
Security and privacy	Encryption-based data protection. Selective information sharing. Fine-grained access. Data confidentiality and integrity. Security-based cloud provider selection.	Efficient data protection algorithms. Security and privacy of fog-based scenarios. Controlled data sharing in multi-providers scenarios. Integrity for multi-provider multi-source computations. Data confidentiality and virtualization.
Data management	Limitations on services for managing metadata. Data management policies and regulations. Managing latency-sensitive data streams.	Metadata management and data traceability. Security and compliance using distributed ledgers. Managing models and data for deep learning.
Networking	High energy consumption. Lack of guaranteed QoS. Multi-tenancy and scalability issues.	SDN-based traffic engineering. Providing a network performance guarantee. AI-based networking.

4) SERVICE MANAGEMENT

There are four fundamental features concerning service management at the gateway of edge computing network, which need to be sustained to assure a trustworthy system [121]. Differentiation, isolation, extensibility, and reliability are fundamental features [122].

- Differentiation

Witnessing the rapid growth in the deployment of the IoT networks, it is predictable that several services will be provided at the network edge, such as smart cities, smart health services, and smart homes each having different priorities depending on their application areas. For instance, time-critical services in particular machine failure alarms,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

fire alarms, and health-related services, such as heart failure and fall detection should be given higher priority than online gaming and entertainment services.

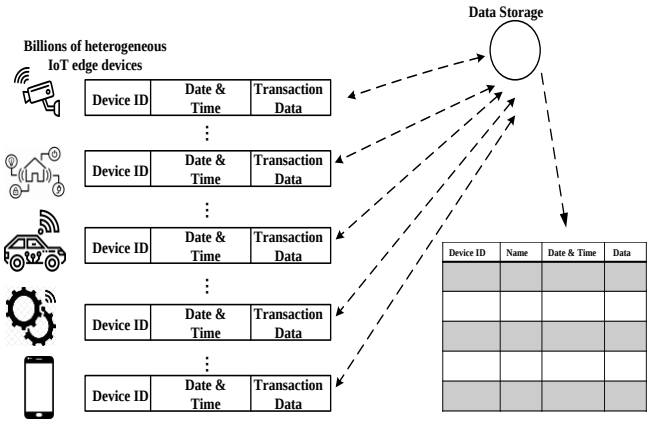

FIGURE 5. Edge computing data abstraction challenges

- *Extensibility*

The edge network operating system should be very flexible to add and drop the things due to wearing out, or a new device purchased by the owner like a mobile communication system. Extensibility could be a massive challenging process for the edge networks due to the dynamic nature of the devices. To solve this problem a complaisant and expandable service managing layer in the edge operating system (EdgeOS) should be designed. This feature which may implicate several dimensions in edge computing is also called scalability. Scalability is defined by four parameters which are described as follows.

Scalable performance: It supports the evolution of fog capabilities in response to QoS demands such as low latency between sending raw sensor data and consequent actuator response.

Scalable capacity: It permits the edge networks to modify the dimension as further applications, edge nodes, users, or things got attached or detached from the network.

Scalable reliability: It allows the system to keep redundant edge capabilities to withstand the faults or overloads. Additional edge nodes can guarantee the integrity and dependability of the distribution at scale, which is a feature of remote access service (RAS). The scalability schemes utilized for the reliability of the edge computing also need to be highly scalable.

Scalable security: It can be attained by the addition of hardware and software modules at the edge gateways as its security requirements e.g. scalable distribution, rights access, cryptography processing capacity, and self-governing security features become more rigorous.

- Isolation

Distributed but coexisting applications and services running over the top of the managed but shared resources at the edge gateway require robust isolation from each other. These services and applications, comprehending various processes of diverse criticality, necessitate strict isolation concerning resource guarantees, fault tolerance, and security. It can be inferred from such necessities that the application performance, the changes, and faults it bumps into do not impact the other application in any way residing in the same edge network. Isolation is going to be another challenge in the edge paradigms at the edge gateway. One more challenge of isolation is how to separate or hide a consumer's private information from the third-party applications. Well-designed access and authentication mechanism need to be integrated into the service management layer in the edgeOS.

- Reliability

Different challenges have been identified herein reliability looking through a diverse account of the system, services, and data. At times it is exceptionally puzzling to ascertain the cause for a service breakdown precisely in the network. It is not adequate to just retain the service during some node failure, but also to provide necessary action after the node crashes add to the purpose of the user.

It is very important for edgeOS to uphold the network topology of the entire system and receive status signals from every node. This feature will allow the system to deploy failure detection, replacement of devices, and quality of data detection services at the edge [123].

Several novel communication protocols have been suggested for the data collection of IoT networks, which serves the purpose well for a large number of sensor nodes and their dynamic circumstances [124]. Though, have much less connection reliability than Bluetooth or Wi-Fi. Thus, a system cannot deliver reliable services with unreliable data sensing and communication protocols.

5) HETEROGENEOUS SYSTEM INTEGRATION

The edge computing paradigm faces several challenges by supporting different types of IoT things having diverse application and service requirements. An edge network is a system made by integrating a combination of numerous platforms, system topologies, and technologies, i.e., it is a heterogeneous system. Hence, for different services running on flexible and heterogeneous platforms, located at different sites are very challenging to program and managing their resources and data.

Unlike cloud computing, edge computing is relatively diverse. Even though various benefits are offered by the distributed network topologies, developers of edge services have to overcome severe challenges in building an application that operates on edge paradigm platforms. Several strategies have been formulated to resolve these issues of the edge [117], [118], [125], but not a single strategy serve the particular purpose. To initialize communication with an edge node, the first step is to discover the type of surrounding edge nodes [21]. Apart from that, numerous server-side codes are necessary to be installed in the edge nodes. Therefore, managing and deploying those server-side programs is also a problematic task.

6) OPTIMIZATION MATRICES

Unlike cloud computing, edge computing has multiple service layers with diverse computational competencies.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

The distribution of workload to each layer becomes a big challenge. The edge professionals need to choose which layer to compute a certain task and the number of processes to assign each layer. Multiple allocation schemes are available in the literature to compute each task according to the computational capability on each layer. The highly complex and extreme cases require high resources to be offloaded to the cloud. Several optimization matrices in this section are argued to find the optimal allocation scheme, comprising latency, energy, bandwidth, and cost.

- Latency

The performance evaluation of edge computing applications is done by calculating the latency matrices, specifically in interacting applications or services [126], [127]. Long communication delays to the cloud and processed response back to the edge node can dramatically influence the real-time or interaction intense services performance. For time-critical services, it is better to process the data in the lowest layer of the edge network having enough computation powers. It is efficient and faster to preprocess the data due to its huge quantity. The nearby possible physical layer is not always available and might be occupied by any other task. We always need to track the resource utilization of the nearest layers to find an ideal layer, so that the unnecessary waiting time is avoided.

- Bandwidth

Looking from the network latency's perspective, the transmission time can be reduced by having large bandwidth, especially for bulky data [121], [128] (video, images, etc.). For short transmission distance, a high bandwidth link can be established to transmit and receive data from the edge. In addition to this, latency can be significantly enhanced by handling the workload at the edge rather than computation done on the cloud. Doing this brings less bandwidth requirement to connect the cloud with the edge node. Moreover, data communication reliability of the network is improved as well, due to the short transmission channel. Although, communication path distance cannot be decreased as the edge cannot always fulfill the computational requirements, at least the data size and bandwidth requirement are significantly reduced due to preprocessing. Therefore, it is apparent to calculate if the high bandwidth is required for the appropriate speed at an edge gateway.

- Energy

At the network edge, the battery is the utmost valuable resource for the devices. At the physical layer, workload offloading to the edge gateway can be considered as an energy-free method [128], [129]. Hence, it is energy efficiency, which decides to offload the whole or a part of the workload to the edge gateway instead of computing locally. A tradeoff between energy consumption by data computation and data broadcasting is the key. In general, the energy characteristics of the workload need to consider first. Is it computational intensive? Resources required doing processing locally? In addition to available bandwidth, the data size, and the network signal strength [130] are few among the parameters which influence the energy transmission overhead [118]. Thus, total power usage must be the accretion of all layered architecture energy costs. Equating with the computation at endpoints, the energy cost will increase drastically due to the increase in the overhead of multi-hop transmission.

- Cost

Consider the service provider's perception, for instance, IBM, Microsoft Azure, Amazon, YouTube, Flipkart, etc., edge computing paradigms offer them less energy consumption and latency, throughput, and end-user experience is increased potentially. Hence, for handling the same workload they can make more profit. For instance, based on most users interests, service providers can place a video or game at the local building layer edge. The higher layers can handle other complex workloads. The service provider's cost is the investment in building layers and maintaining the infrastructure in each layer.

The optimization matrices are diligently interlinked with one another. For the efficient selection of allocation schemes for various workloads, optimization matrices should be the priority. Further, various cost analyses should be performed during execution time. The resource utilization and interference of simultaneous workloads should be maintained and considered as well.

7) PERSISTENT QUALITY-OF-SERVICE AND QUALITY-OF-EXPERIENCE

Quality-of-service (QoS) and quality-of-experience (QoE) are used to evaluate the service quality and quality delivered by the edge system to the end-user respectively [21]. It is important to implement a basic principle of load distribution in edge computing, i.e., to allocate the computational workload among edge nodes according to their available computational resources and capabilities [71], [72]. The nodes should maintain high throughput and reliability when delivering for the primary workload while accommodating the extra workload from the edge device or data center. For instance, the services provided by that particular node might get affected if the base station is overloaded. Datacenter or edge gateway requires comprehensive knowledge about the rush hours of edge nodes in the system to allocate and schedule the workload in an efficient way. The management framework in a network is desirable but monitoring, scheduling, and rescheduling at the infrastructure, platform, and application levels related issues also follow up.

8) STANDARDS AND ECONOMIC REALM

Edge standards can be implemented in reality and made available for users if requirements, interactions, and threats of all the service providers are determined [21]. There are various endeavors to describe an assortment of cloud standards, for example, those by Cloud Standard Customer Council (CSCC), International Telecommunication Union (ITU), International Standard Organization (ISO), IEEE standard Association, and National Institute of Standards and Technologies (NIST). Nevertheless, such models currently need to be reevaluated considering other

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

stakeholders; public and private organizations that own edge data centers, to outline the legal, ethical, and social traits of edge node services.

However, if the performance of the edge is comparable and found reliably benchmarked with the well-defined optimization matrices, then such standards can be implemented. Standard Performance Evaluation Corporation (CPEC) and several other academic researchers are amongst those who took the benchmarking initiative for cloud [131]-[133]. Benchmarking may come across substantial challenges, in the cloud-like noisy environments. The current state-of-the-art researches in the field of edge computing are not yet mature to develop a reliable metric precisely from comprehensive benchmarking suits. Substantial researches are still needed to establish the edge computing standards. Thus edge node benchmarking will be more challenging but offers more research directions. Like the marketplace in cloud computing, a marketplace with diverse edge nodes is feasible in edge computing. Creating such a marketplace will require research in the direction of defining service-level agreements (SLA) and pricing models for edge nodes.

9) ORCHESTRATION AT THE EDGE

Orchestration is a technique to robotically organize, synchronize, and manage multifaceted hardware and software applications and resources. There are various types of potential orchestrations, such as service orchestration (hardware and software service orchestration), infrastructure orchestration (infrastructure orchestration supporting multiple services, consisting of physical or virtual computing, network, and storage resources), and virtual infrastructure manager (interchangeable with resource orchestration) [134]. Some software orchestration platforms are OpenStack, Kubernetes, open network automation platform (ONAP) [135], Cloudify, and Open source management and orchestration (OSM).

Application orchestration management at the edge is the logic describing the sharing of data between the devices and applications to produce commercial intelligence. Installing and handling an application for one IoT device is formulated the same as for a hundred of devices by the edge application orchestration management. The utilization of real-time data is done by taking the cloud traditional systems in an intelligent edge era that is made possible by edge application orchestration.

However, edge computing inflicts some distinctive challenges to orchestration [136] such as mobility, resource constraints, scale, and autonomy. There is a lot of work going on headed for edge orchestration solutions in all open source communities and ETSI. OpenStack, ONAP, and Kubernetes like prominent platforms are presently concentrating on resolving scale challenges for the orchestration components in the core. Ukraine's and StarlingX are other edge computing platforms. But there is no real work concerning lightweight edge orchestrators. Table III shows the requirements of the ONAP orchestration platform to support edge computing.

TABLE III. ONAP ENHANCEMENT AND REQUIREMENTS TO SUPPORT EDGE

Edge computing requirements	ONAP enhancement and requirements (work in progress)
Scalability	To optimally handle enormous edge clouds. To support numerous virtual infrastructure manager (VIM). Hierarchical Federation (Site-level orchestration, fabric control, etc.). Placement decisions across thousands of devices. Discovery of edges and edge capabilities. Statistics gathering agents and database limitations. Auto registration of incoming and outcoming devices.
Security	Joint transport layer security (TLS) on each interaction among ONAP and Edges. Augmentations on the multi-cloud layer and data collection, analytics, and events (DCAE). Certificate registration. Certificate authority (CA) instance that provides certificates for edges; Auto Certificate registration. Certificate private key protection using PKCS11. Software fiddle detection of edges & blacklisting edges.
Regulations	General data protection regulation (GDPR) at the edges is needed for data security. Data encryption needs keys that should not be stored. The key management system (KMS) provides keys on demand. Data placement constraints/regulations.
Performance	Containerized virtual network functions (VNF). Performance evaluation of real-time applications (RTA). Unified networking among VMs and ONAP. Multiple crossing point/interfaces. Efficient resource utilization and high performance.
Edge application provisioning	To create, analyze, and provide MEC status. ONAP has to expose API for application providers. Integration with MEC platforms.
Zero-touch provisioning	Auto edge cloud registration to ONAP. Edge connectivity using private IP addresses. Easy upgradation of the system.
Network slicing	Being addressed by the 5G use case group. Also called physical network function (PNF).
Container, VM and FaaS deployment	Unified networking support from ONAP. Support for FaaS platforms. Network model enhancements to support container and FaaS workloads.
Analytics	Accumulation of statistics & ML analytics. Offer infrastructure data to service providers.

10) SIMULATION PLATFORMS

With the introduction of the concept of edge computing models several new and substantial architecture challenges have been produced for those all involved in the fourth industrial revolution (4IR). Cloud, fog, and edge computing paradigm services and applications contrast by degree and extent of various optimization matrices and factors. Though, the complexity and degree of these factors are the order of magnitude higher than cloud and edge computing paradigms. To test and evaluate the data and computation

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

offloading, resource assignment, and management schemes in edge computing similar to cloud computing. Academicians and researchers are facing several substantial challenges. Primarily, academicians and researchers are not given access to the infrastructure by the commercial service providers, and creating a testbed having a high degree of reliability is equally complicated and expensive, time and resource-demanding. Simulators have been established as the potential techniques to address these challenges. There are a large number of simulators being designed to facilitate the modeling and assessment of diverse characteristics of the IoT systems [137], [138]. All these simulators have been designed without keeping the edge environment needs in mind. Although there are certainly some available simulators supporting edge computing such as CloudSim (iFogSim , EdgeCloudSim, and IoTSim), FogTorch (FogTorch II), OmNeT++ (FogNetSim++) and MiniNet (EmuFog) [139], [140]. Thus, edge computing is in a critical need of simulation platforms addressing the maximum number of characteristics and optimization matrices identified by [140], [141].

11) STANDARD PROTOCOLS

Standardization is a procedure for bringing academicians and leading industries to strive on a single established platform. It is a system made by integrating a combination of numerous platforms, system topologies, and technologies. Hence, for different services working on dynamic and diversified platforms, located at different sites are very challenging to program and manage their resources and data [125]. Therefore, a standardized uncluttered environment is required to be established for the edge to permit smooth and competently assimilation of heterogeneous applications over the edge platforms. The standardization of edge computing is going to speed up the swift growth of edge-based mobile applications through the industry, and eventually upsurge the market size. Mobile edge computing standard characteristics need to be implemented such as computational offloading, data offloading, context-aware information, etc. by standard protocols. Protocols once established can be optimized and improved by researchers and academicians by implementing or modeling in the real platform.

ETSI at the end of 2014 with the establishment of ISG on MEC has started the standardization of edge computing protocols [142], [143]. That is currently the only available international standard in this field of technology, though; new evolving initiatives are on the go in third generation partnership project (3GPP) targeting at the integration of MEC in 5G and beyond technologies. Internet engineering task force (IETF) and other standard developing organizations are as well working from diverse perspectives around edge computing [53]. Thus, a lot of research works are needed towards standard protocol establishment for the realization of edge computing.

12) NETWORK MANAGEMENT

This is the key component of any network and it plays a critical role in fog and edge computing; network management [21] is the technique to interconnect all the smart edge devices to the network and take advantage of all offered resources by several deployed edge nodes. The edge network comprises an enormous number of devices that are spread transversely in huge areas. An attractive function is to accomplish and sustain connectivity. Evolving technologies such as SDN and NFV are considered among the potential solutions to have a remarkable impact on the edge computing network implementation and maintenance. These technologies increase the scalability and decrease the management and maintenance costs of the network [134].

We need a good connectivity mechanism in network management because both mobile and immobile nodes coincide in the network. Here connectivity is the main aspect of the network. There should be an easy mechanism to connect and disconnect the devices from the network to accommodate the uncertainty created by the connecting devices. To increase the edge network with more advanced or smart devices, the intelligent IoT integrator [144] is planning to build a marketplace where all clients can share their data with other different participants and obtain incentives in return.

13) AI FOR EDGE COMPUTING

Recent works in the literature address the AI in a diverse perception of edge computing, IoT, and networks [145]-[158]. However, AI schemes for edge computing consist of eminent models, such as autonomous agents with learning and cognitive abilities, reasoning, prognostic data analytics, and machine learning. Combined contemplation of AI schemes and edge computing, EdgeAI, benefits both technologies in several ways. There are various research challenges and issues that AI implementation in the edge computing faces, thereupon holding back the emergence of AI-based edge computing applications.

- *Training costs of DL models*

It is very challenging to train a deep learning model on resource constraint edge devices. A few endeavors have been made to train models by applying model cropping and quantization but the resulted trained edge models are frequently having lower precision. Thus, the modeling of energy-proficient processes for training neural networks on the edges is a prospective research direction. Therefore, there is a need for building innovative schemes and structures to enhance the speed of training and inference.

- *Heterogeneous Data*

The heterogeneous environment is created in intelligent edge computing systems due to diverse IoT devices available in the market running on variable and dissimilar platforms. To handle the variety of data, ML algorithms require learning information extraction from different data types having different features such as audio, images, video, text, and motion. Learning over multiple modalities (e.g. audio and video) multimodal deep learning schemes are employed [151]. Although these ML algorithms appear potentially promising, in reality, it is challenging to design proper layers for combining features through heterogeneous

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

data and model deployment on the resource-constrained edge devices.

- *Distributed ML challenges*

Researchers have come up with an edge-based distributed algorithm to control the huge quantity of data produced by connected things over distributed computing hierarchies. The hierarchical architecture consists of the end-devices layer (sensors), the edge server layer, and the cloud server layer. Such algorithms deliver sufficiently well accuracy with naturally distributed datasets, for instance, market analysis and fraud detection. Though, the influence of the diversified datasets on the precision and reliability of a distributed system is still an open research direction [152], [153].

The development of the SDN and NFV into 5G and beyond communication systems to govern scattered ML will be an interesting research field for next-generation ML academicians and scientists.

- *Creating New Datasets using Unlabeled Data*

The capability to train with unknown input data is an indispensable characteristic of the deep learning algorithms. For creating new datasets, the enormous unlabeled data created by the end devices are the best sources, however advanced ML algorithms need to build datasets with less noisy labels.

Data augmentation of edge and other end devices is another active research field to enrich the deep learning performance. Overfitting problems in ML models are evaded by producing sufficient data in augmentation [157], [148].

- *Accuracy of the Learning Model*

It has been studied and established that edge-based deep learning models have applications in time-critical services like health care [156], [158], but smart and intelligent systems are required to possess a great level of accuracy due to safety-critical aspects before using in the health care sector. Unavailability of the processed datasets is another challenge.

- *Augmented Cognition*

Currently, research is going on to explore in what way DL and edge paradigm can be utilized to augment human cognition to build an adaptive human-machine alliance by guiding the human for unacquainted workloads and for magnifying the memory capability of humans [155], [156]. Such techniques can transmute the abilities to execute both routine and complex jobs by humans having low cognitive abilities, but challenges about privacy, security, accuracy, and ethics necessitate to be facilitated before deployment of such systems.

14) SECURITY AND PRIVACY

User secrecy and data confidentiality guarantees are extremely significant security functions that ought to be given at the network edge. Taking use case of a smart home set up with IoT, the data generated from it can be exploited to extract a large amount of private information. In this situation, providing the services for the users without exposing their private information is a big research challenge. One way is to remove the private data before computing at the cloud servers. Thus, deploying the processing capabilities at the edge of the smart home is the better method to ensure user secrecy and data confidentiality. In ensuring data and user secrecy at the network edge, there are still several challenges. Fog and edge computing are well-thought-out as the promising augmentation of the distributed computing standard. For this reason, several cloud applications accept the visualization of fog and edge computing by deploying computational resources near the network edge.

To ensure the security and privacy on edge and fog devices in a network, researchers have to assure the most important characteristics such as confidentiality, integrity, and accessibility [2]. Among these characteristics, confidentiality and integrity provide data privacy guarantee while accessibility assures the sharing of the resource between edge nodes whenever required [159]. Authentication, access control, privacy, and intrusion attack are the main security concerns in fog and edge computing. Whenever any edge device is connecting or leaving the network without any restriction in a dynamic IoT network, a connectivity structure has to confirm that the security is well-preserved at the edge and fog node by authenticating the different devices in the network [160].

An edge environment is made out of heterogeneous devices distributed in a multi-layer model with their security issues. Besides, new security issues appear from consolidating these devices to frame another IoT ecosystem. In Table IV we have discussed all conceivable security issues of the edge computing system considering each architectural component. Analyses of all the above-mentioned attacks that can target any network infrastructure are all discussed in [161].

IV. CLASSIFICATION OF SECURITY THREAT MODELS IN EDGE COMPUTING

A. DISTRIBUTED DENIAL-OF-SERVICE ATTACKS

Due to the high mobility of the user devices, the edge computing security threat situation is persistently evolving. Attackers can join the network group easily and it is hard to plan and implement the new security mechanisms due to heterogeneous devices from different device suppliers using different technologies [161]. Most of the security threats and attacks encountered by edge computing and related paradigms are mainly due to design faults, bad configuration, and implementation bugs. By the security attacks and threats that edge computing is facing, the attacker's main focus is to introduce the design errors, misconfigurations, and bugs enactment. Edge servers are computationally less powerful software and hardware relative to the cloud to withhold strong defense mechanisms against the attacks [3] and are more prone to distributed denial-of-service (DDoS) attacks.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

TABLE IV. EDGE COMPUTING THREAT MODEL

Edge components / Security issues	Network infrastructure	Service infrastructure (data)	Service infrastructure (core)	Virtualization infrastructure	User devices
Denial-of-service (DoS)	Yes	No	No	Yes	No
Man-in the-middle	Yes	No	No	No	No
Rogue component	Yes	Yes	Yes	No	No
Physical damage	No	Yes	No	No	No
Privacy leakage	No	Yes	Yes	Yes	No
Privilege escalation	No	Yes	No	Yes	No
Service or VM manipulation	No	Yes	Yes	Yes	No
Misuses of resources	No	No	No	Yes	Yes
Injection of information	No	No	No	No	Yes

Attack Specifications: The DDoS attacks happen when edge devices are communicating with edge servers, here attackers first attack cluster edge devices to take full control over them; then it tries to control every device to launch DDoS attacks and target the edge servers and shut down the services of the whole network. A simple architecture of the DoS attack is shown in Figure 6 to understand the DoS attack.

Current Defense Solutions: The root causes of the above mentioned two attacks are given as:

a) Protocol-level design vulnerabilities within the network communication protocols are the main root cause of flood-based attacks.

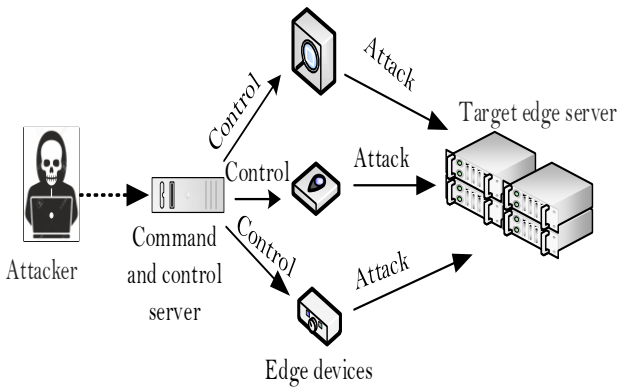

FIGURE 6. The architecture of the DDoS attack

b) While in the code-level vulnerabilities which also trigger memory failures are the cause of the zero-day attacks. The current defense solution mainly adopts a detect-filter philosophy in flooding-based attacks while code-level vulnerability identification is mainly focused on zero-day attacks.

1) FLOOD BASED ATTACKS

Flooding based attack is a type of DoS attack in which the attacker's main aim is to shut down the normal services of servers by sending a large amount of flooded malicious network packets, which are mainly classified as internet control message protocol (ICMP), user datagram protocol (UDP), SYN flooding (synchronous) and Slowloris. As per the UDP flooding attack, the attacker continuously sends a huge number of noisy UDP packets to start targeting edge servers to make servers incapable to handle and interrupt the normal function of edge servers [162].

In ICMP flooding attack the attacker tries to exploit the protocol of ICMP to craft an attack by sending an unlimited number of echo request packets to target edge server as soon as possible without waiting for their reply, resulting in the significant system-wide slowdown [163]. The transmission control protocol (TCP) also initiates a huge number of SYN requests to target edge server with a spoofed IP address, while the server waits for ACK confirmation which never comes [164].

Defense solutions against flooding-based attacks**:** In flooding-based DDoS attacks the detection can be mainly classified into two categories as defined below.

- *Per-packet Based Detection*

Its main aim is to find a flooding-based attack at the packet level, this type of attack happens by sending an enormous number of malicious network packets, filtering, and detecting these can have an effective defense. This scrutiny was investigated in [165], which recommended adding a packet filtering mechanism into congestion control frameworks to reduce the attacks. In [166], the authors tried to develop a more effective scheme and presented two new mechanisms to find DDoS attacks by evaluating the distance values and traffic rates. Moreover, [167] used congestion control in IP networks for spotting possible DDoS attacks based on packet identifiers.

- *Statistics-based detection*

This approach, however, finds DDoS attacks based on the initiation of clusters of DDoS traffics. These methods do not require the per-packet information and IP/MAC addresses for attack detection. The available statistical detection techniques use either machine learning tools or packet entropy. Various entropy mechanisms have been developed by researchers in [168], [169] to find the best possible DDoS traffic. These methods also require somehow manual efforts which is a bit challenging if DDoS traffics is encrypted. But to find accurate detection this also

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

requires the distribution of a bulky quantity of traffic. To detect encrypted attacks, a deep learning model using auto-encoder is proposed in [170]. In an SDN [171], DDoS attacks are identified by using Neural Networks.

2) ZERO-DAY DDOS ATTACKS

In this attack, the attacker must find an unknown vulnerability called a zero-day vulnerability, in a piece of codes running on the edge server which is in target. This type of attack may cause memory corruption and can even lead to the crushing of services on that very edge. These types of attacks are very difficult to defend since it exploits a zero-day vulnerability that is almost unable to detect by the user [161].

Defense solutions against zero-day attacks: While defending zero-day attacks a new mechanism was developed by researchers such as ECC-memory and point twistedness detection in [161], [172] to advert probable memory drips in the program, but still these techniques cannot work without the source codes which are not present at edge devices. The authors of [173], [174] proposed different approaches based only on the firmware to perform memory analysis while the authors of [161], [175] used deep learning methods to find the desired solutions, for example, graph neural networks (GNN), natural language processing (NLP), and recurrent neural networks (RNN) which can identify weaknesses in firmware with high correctness rates. The authors of [176] used the SDN to establish an IoT firewall to minimize the attack surface of exposed IoT devices.

B. SIDE-CHANNEL ATTACKS AND DEFENSE MECHANISM

The side-channel attack is the type of attack where the weaknesses in the implemented algorithm are not exploited but are based on sensitive information gained from the execution of certain ML or DL models on the data obtained from side-channels [161]. Communication signals, power consumption, and smart edge devices (embedded sensors) are the most common edge computing side channels. The attacker continuously monitors and exploits the communicated data such as communication signals, power consumption data, etc., to mine some sensitive information, and meanwhile, edge device communication channels are covertly accessed by attackers to steal the desired data generated by the embedded sensors publicly available. Table V elaborates on the different side-channel attacks with their behavior and defending mechanism.

TABLE V. SIDE-CHANNEL ATTACKS AND THEIR BEHAVIOR

Attack type	Physical security attack	Behavior	Defending mechanism
Chance based attack	Non-invasive, passive attack	The attacker exploits the application behavior of cache to extract data related to the encryption algorithm	Sandbox (dynamic binary translation)
Timing attack	Non-invasive, passive attack	Attacker continuously monitors and analyzes the time taken by system right from data input to encrypted data output and determines algorithm strength	Stop Watch, Attacker VM
Power monitoring attack	Non-invasive, passive attack	From monitoring the device power consumption, the attacker extracts the sensitive key details	Game theory approach
Electromagnetic attack	Non-invasive, passive attack	Electromagnetic radiations produced by the system are monitored and certain signal processing is applied to get the desired information	To design an advanced encryption standard (AES) cryptographic circuit while using ROM
Acoustic cryptanalysis	Non-invasive, passive attack	This examines the audio sound produced by PCs and other gadgets including the keypress	Secure vibe and sensor are used
Differential fault analysis	Non-invasive, passive attack	The attacker introduces the various faults and errors in a particular system and then monitors the incoming data from the network.	New AES and silicon-level countermeasures
Data reminisce	Invasive, passive attack	The attacker tries to recover the deleted data from the system's primary memory.	To erase the data from SRAM safe circuits are used
Optical attack	Non-invasive, passive attack	The penetrating data will be hacked by the visual recording technique.	Public key cryptography

V. BLOCKCHAIN TECHNOLOGY

A. INTRODUCTION TO BLOCKCHAIN TECHNOLOGY

The blockchain is a distributed ledger open to anybody. Fundamentally, it is a decentralized, pooled, and hard to tamper database of records or ledger of every single record (transactions) that has ever been executed and communicated between the parties across a peer-to-peer (P2P) network [177-180]. It comprises data blocks chained together with each dependent hash values that have verified or timestamped using a consensus mechanism by the majority of miners actively involved in authenticating and verifying transactions. Once the transactions are

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

timestamped and verified by miners, they can never be erased or tampered (i.e. immutable).

The data authenticity and integrity proof is provided by the integration of elliptical-curve cryptography (ECC) and the secure hashing algorithm (SHA-256). Blockchain is a distributed network designed to make non-trusting nodes communicate with each other in a secured manner without any trusted central control authority [42-44]. Key characteristics of blockchain are listed and described in Table VI below. Use cases of blockchain are growing to several fields for instance banking, health, IoT, BOINC (Berkeley open infrastructure network computing grid), voting systems, and many others [31-33]. The companies and groups currently working on blockchain 3.0 projects are IOTA [34], Dfinity [181], EOS [182], Lightning Group [138], Hash Group [183], NEO [62], and Ethurium itself [35]. These projects seek to extend the capacities of current blockchains. The structure of a block in the blockchain is shown in Figure 7.

With the rise of the blockchain technology, smart contracts fundamentally containers of codes having self-verifying, self-executing, and immutable properties have turned out to be the most sought-after technologies [42], [184]. A vital premise for smart contracts is to characterize an obligatory agreement between the participating parties and make sure every party carries out their obligations under the agreement. Smart contracts remove the need for any legal centralized control authority between the contract members and also support transactions between untrusted users without any third-party dependency, the necessity of direct mutual communication between two parties [185], and altogether make the system robust against any malicious attacks. The structure of a smart contract consisting of value, functions, state, and address [186]. Smart contracts in a blockchain network are nothing more than the functions placed on the blockchain having the capability to implement them. Unique addresses allocated to trusted entities by the blockchain can be utilized to initiate a transaction to a smart contract. There are three roughly categorized types of blockchains: the restricted network (limited to a certain group of users), unrestricted network (open to anyone to join in), consortium networking, (only certain selected nodes can validate the users) or the hybrid blockchain (restricted and unrestricted blockchain). Table VII lists distinctions between different blockchain networks based on certain key characteristics.

TABLE VI. KEY CHARACTERISTICS OF THE BLOCKCHAIN TECHNOLOGY

Characteristics	Description
Decentralization	Blockchain does not need to validate each transaction by a centralized trusted agency. The transaction information is stored, verified, and updated distributedly. Diminishes server costs and mitigates bottlenecks at the central server.
Transparent	Data recorded on the blockchain network system are transparent to each member, data can be updated upon getting verified by all nodes and thus can be trusted.
Open Source	The blockchain technology is an open-source platform, that can be used by anybody to build their applications without taking permission from any regulatory authority.
Persistency	Since the majority of miners are actively involved in authenticating and verifying transactions across the network, it is nearly immutable. To change data of transactions, the attacker needs to control 51% of the participating members in the network.
Auditability	Transactions are tagged with timestamps, so that any user may find it using the public key in the blockchain. The auditor can trace all the timestamped transactions to verify the transactions. Stored data transparency, trust between nodes, and traceability are improved by auditability.
Anonymity	A generated address is used by each node to communicate and work in a blockchain network. Conserving privacy on the transactions is encompassed in the blockchain.

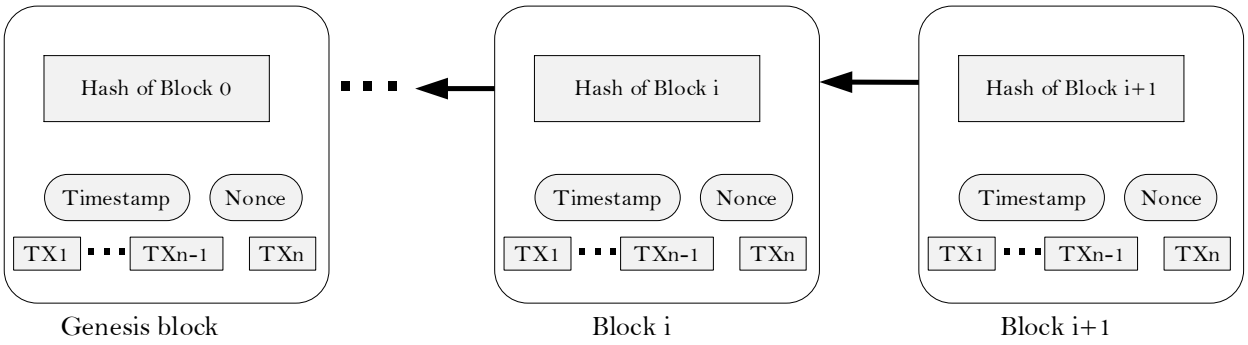

FIGURE 7. Structure of a block in a blockchain

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

B. OVERVIEW OF BLOCKCHAIN TECHNOLOGY ARCHITECTURE

Blockchain is a structure of blocks holding a comprehensive list of transaction history identical to a standard public ledger. The blocks in the network are certified by consensus between the participating edge nodes employing a digital signature based asymmetric cryptography mechanism. The blockchain architecture is divided into four layers: applications, distributed computing, platform, and infrastructure. All the hardware modules are in the architecture layer like network facilities, data storage, and nodes. Remote procedure call (RPC), API, and representational state transfer (REST) are the functions facilitated by the platform layer. Table VIII describes the types of nodes in the blockchain network [35], [184]. The validated block consists of a timestamp, parent block's hash value, and a nonce for the verification of the hash. The integrity of blockchain is ensured through the genesis block (first block). Genesis block contains the entire hash function, access control rights, the block generation interval, and the block size. All other blocks connected in the chain are linked to this block. The block components used in blockchain transaction execution are shown in Figure 8 and described in Table IX.

There is a surge in the development of blockchain platforms in the last decade has acquainted with different architectures based on application requirements in a budding cooperative environment [47]. The literature has classified the blockchain architectures based on their characteristics as single-ledger based architecture [35], [47], multi-ledger-based architecture [185], [186], and interoperability-based architecture [187], [188]. In a single-ledger based system, the corresponding architectures vary depending on network applications such as public [189], private [190], and hybrid blockchains [50].

Interoperability increases the security of a blockchain network by connecting it with a different blockchain. The anchoring process is used to connect two blockchains.

TABLE VII. COMPARISON OF THE CATEGORIZED BLOCKCHAIN NETWORKS

Characteristics	Restricted network	Unrestricted network	Consortium/hybrid network
Consensus determination	Certain group or an organization	All connected minors	The certain selected set of miners
Immutability	Possible to get tampered	Impossible to tamper	Possible to get tampered
Efficiency	High	Low	High
Centralization	Yes	No	Partial
Consensus process	Permissioned	Permissionless	Permissioned
Read permission	Could be anyone (private or public)	Public	Could be anyone (private or public)
Transaction approval frequency	Short (in milliseconds)	Long (more than 10 minutes)	Short (in milliseconds)
Unique selling proposition (USP)	Cost-cutting (reduces transaction costs)	Disruptive	Cost-cutting (reduces transaction costs)
Consensus mechanism	Voting or multiparty consensus Enable finality	Proof of Work Proof of Stake, etc. No finality	Voting or multiparty consensus Enable finality
Energy utilization	Low	High	Low
Speed	Fast	Slow	Fast
Complexity	Low	High	Low

TABLE VIII. CLASSIFICATION OF BLOCKCHAIN NETWORK NODES

Node Type	Narrative
Miner	Special nodes having maximum computational power. It is responsible for adding blocks for transactions and running proof-of-work consensus.
Full node	It has sufficient storage and computation power. It stores a copy of the whole blockchain. It continuously authenticates the integrity of all transactions, creating decentralized and truthless infrastructure.
Thin client	The thin client requires minimum storage and computation power. It only saves headers that enclose hashes of the transactions contained by the blocks.
Server-trusting client	To create secure and lightweight clients for resource scare systems, a bitcoin client API (BCCAPI) is proposed. No transaction can be created by the servers without the client's approval as it contains only the client's public key.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

Anchoring guarantees the prodigious immutability of the sidechain. Figure 9 represents all the architectural components in public, private, single-ledger, multiple-ledger-based blockchain platforms based on their characteristics.

C. DECENTRALIZED CONSENSUS PROCESS

In blockchain applications, designers need to resolve the double-spending problem and byzantine generals problem. In a blockchain, there are no centralized transaction systems and thus are solved together by several distributed nodes by verifying and validating the transaction. Though, due to the mischievous attacks on some nodes to tamper the communication data. The other normal nodes first need to find the tampered data and validate the consistent results from the other network nodes. Thus, to ensure the trustworthiness and regularity of the data and transactions, distributed consensus mechanism [29-30], [191] is adopted by the blockchain [192]. To reach the consensus between the nodes in a blockchain [185], different approaches with their mechanisms are presented in Table X.

Several consensus algorithms have been developed in the last three decades for blockchain. In the case of employing blockchain in edge computing and IoT, it is essential to first understand the different blockchain platforms, consensus algorithms, and their current constraints. Table XI differentiates different consensus algorithms based on their characteristics.

D. PLATFORMS FOR BLOCKCHAIN SIMULATION

By demonstrating its potential and capabilities for diverse applications, blockchain technology has attracted the attention of researchers and industry since the decade. These features paved way for much technological advancement. Several platforms have evolved using blockchain technology; most of these platforms were developed for financial applications like Bitcoin. These platforms help individuals and companies in creating applications based on the blockchain technology without any need to develop their blockchain. Blockchain is now finding its applications across diverse fields such as healthcare; IoT, supply chain, data management, and security and privacy are the new trends and evolving areas of research.

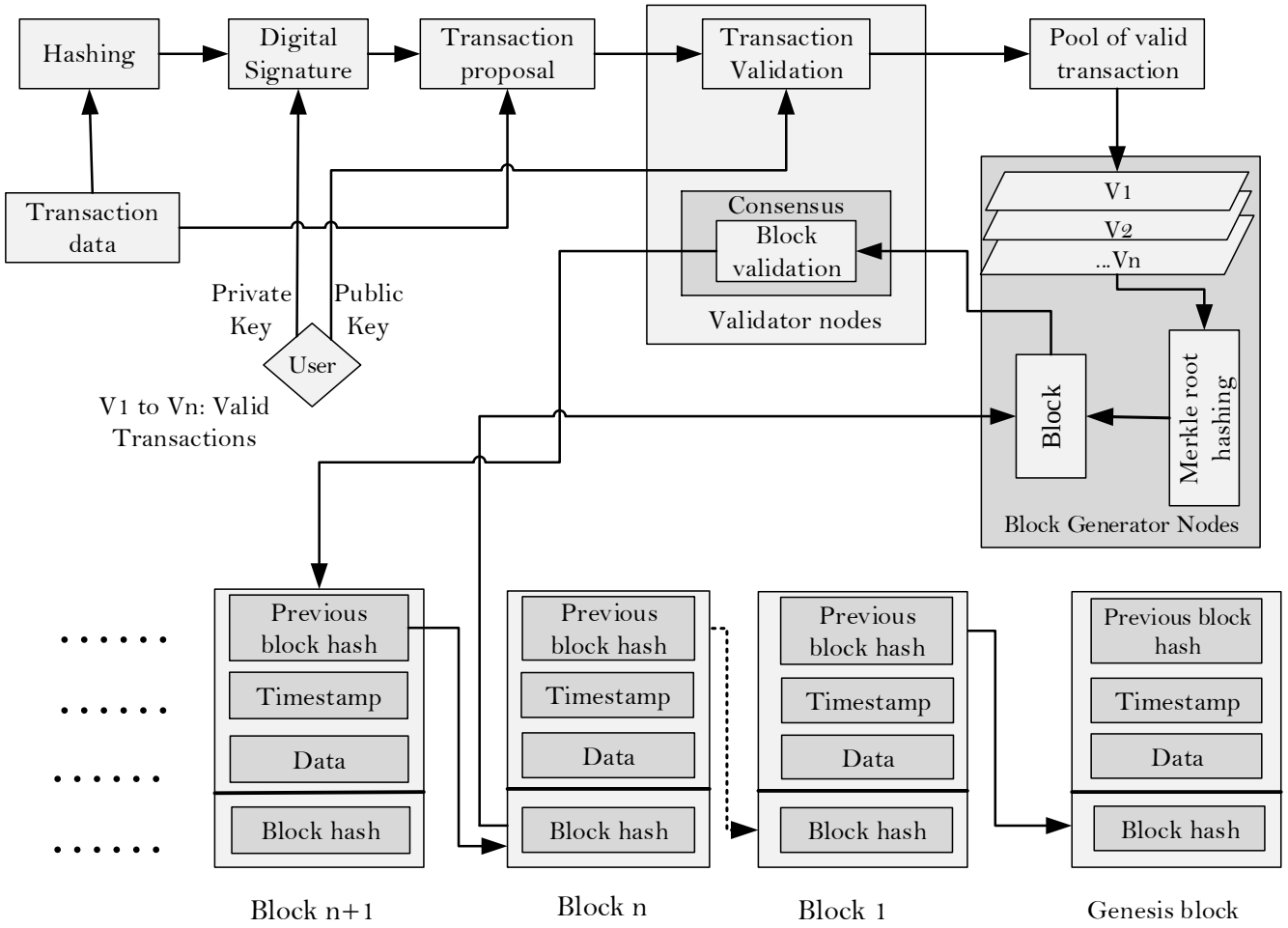

FIGURE 8. Execution of blockchain transaction

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

TABLE IX. TRANSACTION EXECUTION FLOW COMPONENTS USED IN A BLOCKCHAIN

Component	Depiction
Transaction	Process or event used to update the data stored in the blockchain ledger. It may contain payment information, data, execution of smart contracts, or programs stored depending upon the application.
Block	Block is a file containing valid transaction information in the network such as date, time, and the value of the transaction. It also stores data about participants and distinguishes between blocks.
Merkle tree	A hashing algorithm is used to calculate the hash of each transaction individually and then multiplex to get a single hash called Merkle tree hash value.
Block hash	Computed by hashing the header of the block twice and is unique for each block.
Previous block hash	The Hash calculated in the parent block is used to compute the hash of the current block. It helps to make the blockchain ledger immutable.
Timestamp	It is used to mark the block creation time with the block.
Block version	It specifies the form of the blockchain protocol used.
Mining	A method to validate and secure the transactions in the blockchain system in a decentralized manner.
Genesis block	It is the first block in the blockchain ledger to which all other blocks are connected. It contains all the network configuration features, the consensus algorithms to be implemented, the hash function, access control rights, the block generation interval, and the size of the block.

TABLE X. DIFFERENT CONSENSUS APPROACHES AND THEIR MECHANISM IN BLOCKCHAIN

Consensus approach	Description
Proof of Work (PoW)	PoW is a complex consensus mechanism of authentication. The newly generated block is broadcasted in the network for authentication and gets attached to the chain only if its calculated hash is equal to a certain predefined value.
Proof of Stack	It is an energy-efficient form of PoW. Users prove their ownership by the number of previous transactions and reduce the chances of a malicious attack.
PBFT	Practical Byzantine Fault Tolerance (PBFT) is a less complex, multi-stage algorithm enhancing its efficiency by addressing transmission errors but needs exponential operations for the Byzantine system. A transaction needs 2/3 votes from participating nodes to enter the next state.
DPoS	In differential-PoS (DPoS) a small number of elected participants validate the blocks quickly and malicious nodes are easily detected. It is highly efficient and has less power consumption than PoW and PoS.
Ripple	It consists of server and client categories of nodes divided into subnetworks and their mutual trust is used for consensus. Each server node is provided with a unique node list (UNL) to receive the validity or authenticate the transactions from that list.
Tendermint	Comparable to PBFT except tendermint nodes need to lock their coins to develop as authenticators. Tendermint process is distributed into three stages namely the prevote phase, pre-commit step, and commit step.
Raft	A CFT-based (crash fault tolerance) consensus algorithm mostly containing five server nodes and is robust to two-node failures simultaneously. The server node is in one of the subsequent states: leader, follower, or candidate.

Numerous blockchain platforms [193] have been developed and several are evolving targeting different applications like IOTA, NEO, EOS, etc. It is a critical step to decide the most appropriate platform to start designing and building any blockchain project due to a lot of technical characteristics that need to be evaluated for a specific project. The technical characteristics of several blockchain platforms have been listed in Table XII and Table XIII for

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

selecting an appropriate off-the-shelf blockchain platform for a particular application.

VI. CHALLENGES AND RISKS OF BLOCKCHAIN TECHNOLOGY

For the forthcoming generation of systems based on internet interaction blockchain technology has come up as a highly promising technology, such as IoT, edge computing, smart contracts, and security services. Even though the blockchain technology has evident potential for building the next generation internet systems, it has its technical challenges that need to be addressed. Some key challenges of blockchain technology are described as follows.

A. PERFORMANCE AND STORAGE CAPACITY

Expanding the blockchains can harm the throughput and redundancy as the system necessities to manage the expanded volume of data exchange and processing. Using PoW which is a central processing unit that helps to mine faster is also expensive and energy-consuming. In PoW protocol there is also the risk of numerous branches that can approach towards double-spending problem [42].

Meanwhile, with the growing size of the network, the nodes require more and more resources. The storage requirements are significant at nodes of the full chain that shrinks the capacity scale of the network, and the performance also gets decreased due to oversized chain.

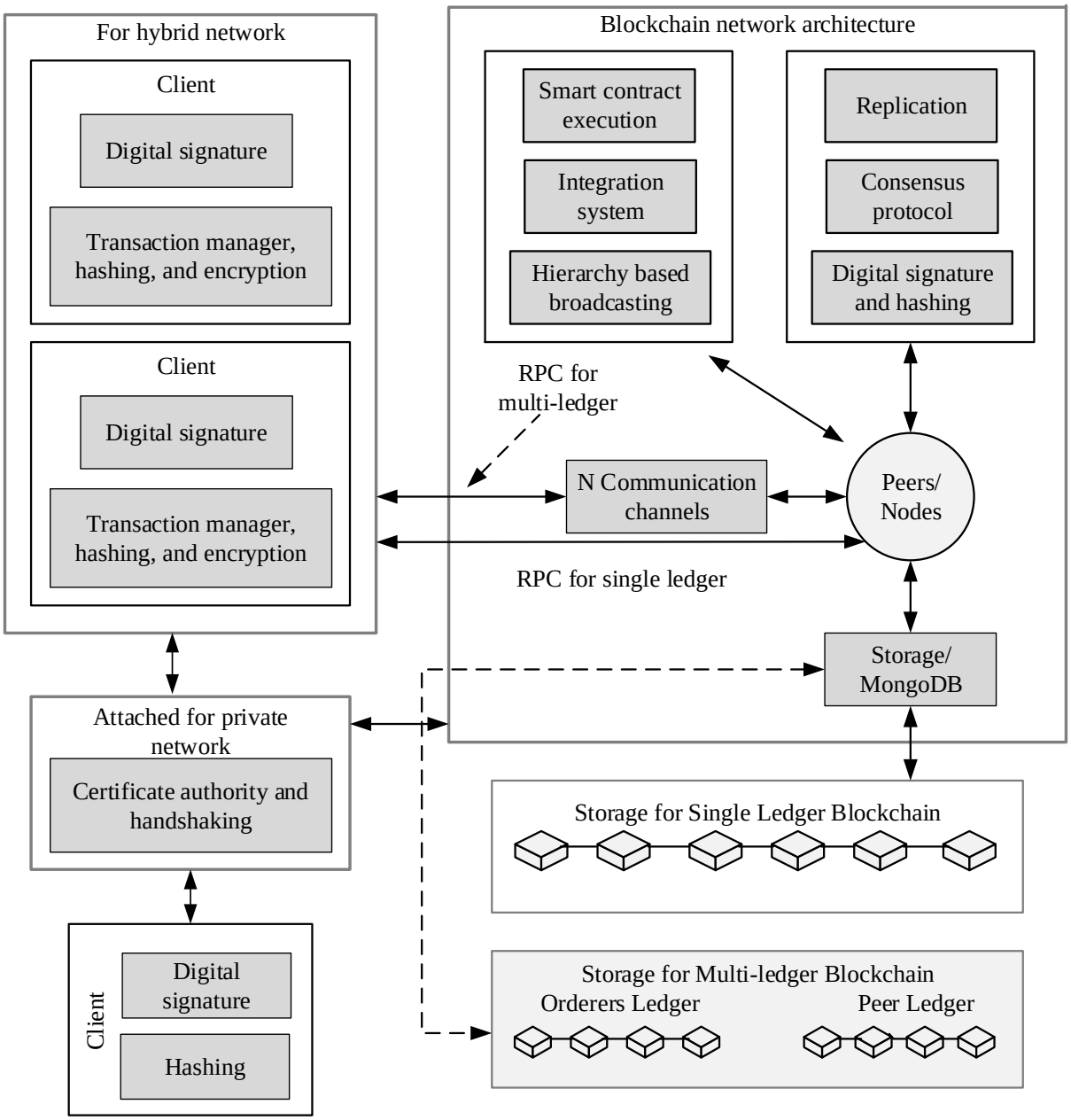

FIGURE 9. The architectural components of different blockchain platforms

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

TABLE XI. CHARACTERISTIC COMPARISON OF CONSENSUS ALGORITHMS

Characteristics	Consensus algorithms								
	PoW	PoS	DPoS	PBFT	Raft	Ripple	Tindermint	POET	POC
Byzantine fault tolerance	50%	50%	50%	33%	N/A	20%	33%	28.5	NA
Crash fault tolerance	50%	50%	50%	33%	50%	25%	33%	NA	NA
Verification speed	>100s	<100s	<100s	<10s	<10s	<1s	<10s	<10s	<100s
Throughput (TPS)	Low	Medium	Medium	High	High	High	High	Medium	Low
Scalability	Low	Strong	Strong	Weak	Weak	Strong	Medium	High	Medium
Energy saving	No	Partial	Partial	Yes	No	Yes	Yes	Yes	Partial
Node identity management	Open	Open	Open	7 PREM	7 PREM	7 PREM	7 PREM	Open	Open
Blockchain type	Both	Both	Both	Private	Private	Private	Private	Both	Public
Transaction finality	𝓘 Prob.	𝓘 Prob.	𝓘 Prob.	Immediate	𝓘 Prob.	𝓘 Prob.	Immediate	𝓘 Prob.	Immediate
Tolerated power of adversary	<25% 𝒈 Comp. power	<51% stake	<51% validators	<33.3% faulty replicas	<40% faulty nodes	<20% faulty nodes in UNL	<33.3% byzantine voting power	<25% 𝒈 Comp. power	<25% 𝒈 Comp. power
Trust model	Untrusted	Untrusted	Trusted	Semi-trusted	Untrusted	Trusted	Trusted	Untrusted	Untrusted
Example	Bitcoin	PeerCoin	Bitshares	Hyper-ledger	Quorum	Bitcoin	Bitcoin	PeerCoin	Bitshares

7 PERM: Permissioned 𝓘 Prob.: Probabilistic 𝒈 Comp.: Computing

B. SCALABILITY

The blockchain becomes bulky with every passing day by increasing the number of transactions and has surpassed 100 GB storage at present. For validation, all these transactions need to be stored by the nodes. Moreover, there is a restriction on the block size and a minimum time interval gap between block generations. Any conventional platform needs thousands of transactions per second (TPS) whereas the bitcoin processes only 7 TPS that makes it impossible to execute millions of transactions in a real-time scenario. Even due to the small capacity of blocks a large number of transactions might be delayed and then goes for those transaction executions with a high service charge. Though, with a large block size, the data propagation speed is decreased and leads to blockchain branches. This scalability problem has been proposed by many authors in two categories: Storage optimization of blockchain and redesigning blockchains [194], [62]. The transaction speed (in TPS) and the time interval to generate a new block describe the computational power and have a direct effect on the confirmation time of a transaction. Therefore, the scalability issue is challenging.

C. PRIVACY AND SECURITY

Whenever a customer makes a transaction the blockchain will provide privacy and secrecy to the personal data of the user. Blockchain technology will not use your real identity, whereas it will generate addresses for each user. The users can do their transactions by using public and private keys without exposing their real identity. However, because of the visibility of publicly public keys, transaction privacy cannot be guaranteed by blockchain technology [41], [47], [195] described by the author [191] in detail.

In bitcoin, among many vulnerabilities and security threats that are discovered, 51% attacks [65] are common attacks, and double-spend attacks are also possible in fast payments in blockchain [43]. Similarly, in these scenarios race attack is also possible. There are many such attacks in blockchain technology for instance DoS, a man in the middle, Finney attack which is a more erudite double attack, and eclipse attack. Those attacks can damage the inter-edge node communication channels.

D. KEY MANAGEMENT

A blockchain-based edge computing solution must solve the issue of key management in edge devices [196]. If a system requires edge devices to connect and transfer digital currency from one wallet to another in a truly autonomous manner, where will the private keys be stored and managed? Will devices generate and hold the keys? Solutions are being developed to address this issue. However, there is still much concern about securing keys properly for edge devices.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

TABLE XII. CHARACTERISTIC TABLE OF DIFFERENT BLOCKCHAIN PLATFORMS (PART 1)

Platform	Bitcoin	Ethereum	HyperLedger	Ripple	SBFT	Stellar	Eris-DB	Dfinity
Consensus	PoW	PoW, PoS	PBFT	POS, Ripple Consensus Lsdger	SBFT	Stellar Consensus Protocol, TE/PoP	Tendermint, BFT, PoS	Blockchain, Nervous System, PoS, PoW
Participating nodes in consensus	High 11.5 K	High	High	Low 55	Low 5	Low 20-30	NA	Millions of nodes
Data model	Transt-based	Account based	Account-based	UTXO-based	Account-based	Account - based	Account-based	Account-based
Smart contract execution	NA	Ethereum Virtual Machine (EVM)	Dockers	NA	Haskell Execution, EVM	Dockers	EVM, Dockers	EVM
Language	Forth	Solidity, Serpent, Low-level Lisp (LLL)	Golang, Java	C++, JavaScript	Pact, C++	Javascript, Golang, Java, Ruby, Python	Solidity,	Solidity, Serpent, LLL
Fully developed	Yes	Yes	Yes	Yes	Yes	Yes	Transition	Yes
Trustless operation	Yes	Yes	Trusted validator nodes	Trusted validating servers	Trusted	Trusted	Trusted	Trusted
TX throughput	7 TPS	20 TPS	Can achieve upto 1000	1500 TPS	50-70 TPS	420 TPS	NA	1k – 3k TPS
Miner participation	Public	Both	Private	Private	Both	Private	Public	Private
Scalable	No	No	Yes	Yes	Yes	Yes	Yes	Yes
Multiple applications	Financial	Yes	Yes	Financial	Yes	Financial	Yes	Yes
Consensus finality	No	No	Yes	Yes	No	No	Yes	Yes
Smart contracts	No	Yes	Yes	No	Yes	Yes	Yes	Yes
TX integrity and authenticity	Yes	Yes	Yes	Yes	Yes	Yes	NA	Yes
Data confidentiality	No	No	Yes	Yes	Yes	Yes	Yes	Yes
ID management	No	No	Yes	Yes	Yes	Yes	NA	Yes
Key management	No	No	Yes	Yes	Yes	Yes	Yes	Yes
User Authenticity	Digital signature	Digital signature	Based on the enrollment certificate	Digital signature	Digital signature	Digital signature	Yes	Yes Threshold signature
Device authentication	No	No	No	Yes	No	Yes	No	Yes
Attack probablity	51% linking attack	51%	>1/3 faulty nodes	> 1/4 faulty nodes	51%	1/3 malicious nodes	25%	67%
Latency	10 mins	15-20 secs	<1secs	4 secs	<1 secs	5 secs	Low	5 secs
Scalability of a peer network	High	Low	NA	High	NA	High	Yes	Yes
Ledge type	Open	Open	✗ PERM	✗ PERM	Both	Open	✗ PERM	✗ PERM

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

TABLE XIII. CHARACTERISTIC TABLE OF DIFFERENT BLOCKCHAIN PLATFORMS (PART 2)

Platform	Parity	Tezos	Cordano	Hyperledger sawtooth	IOTA	EOS	NEO	Hyperledgher
Consensus	PoA	DPoS	Raft	PoET	Tip Selection Algo.	DPOS	DBFT	Pluggable consensus
Participating nodes in consensus	NA	Low	Low	Hgh	High 250	Low 21	Low 7-100	Low
Data model	Account-based	Account-based	UTXO-based	Account-based	Account-based	NA	NA	Account-based
Smart contract execution	EVM	Dockers	JVM	TEEcorda protocol	NA	Block producers	NeoContract (NeoVM)	EVM,
Language	Solidity, Serpent, LLL	Tezos contract script language	Kotlin, Java	Python	javascript, Python, C#, Java, Golang	C, C++	C#, Java, JavaScript, Python, Ruby	Java, Go, Solidity, JavaScript
Fully developed	NA	No	NA	Transition	Transition	Yes	Yes	Yes
Trustless operation	Trusted	Yes	Trusted	Trusted	Yes	Yes	Yes	Trusted
TX throughput	80 TPS	30-40 TPS	2577 TPS	1000 TPS	500 TPS	50K TPS	1000-10K TPS	>2500 TPS
Miner participation	Private	Public	Public	Private	Both	Private	Public	Private
Scalable	Yes	Yes	Yes	High	Yes	Yes	Yes	Yes
Multiple applications	Yes	Yes	Yes	Yes	Financial	Yes	Yes	Yes
Consensus finality	Yes	No	Yes	Yes	No	Yes	Yes	Yes
Smart contracts	Yes	Yes	Yes	Yes	Currently No	Yes	Yes	Yes
TX integrity and authenticity	Yes	Yes	NA	Yes	Yes	Yes	Yes	Yes
Data confidentiality	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes
ID management	No	Yes	Yes	Yes	No	Yes	Yes	Yes
Key management	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes
User authenticity	Yes	Yes	Yes Notary	Yes	Digital signature	Yes	Yes	Yes
Device authentication	Yes	Yes	No	No	No	No	No	Yes
Attack probablity	50%	51%	< 40% faulty nodes	< 25% of Comp. power	2/3 vulnerable nodes	> 1/3 faulty nodes	1/3 faulty nodes	NA
Latency	>10 secs	60 secs	3-5 mins	<10 secs	NA	1.5 secs	15-20 secs	<1 sec
Scalability of peer networks	Yes	Yes	Low	High	High	Low	Low	High
Ledge type	✗ PERM	✗ PERM	✗ PERM	Both	Both	✗ PERM	✗ PERM	✗ PERM

E. ENERGY CONSUMPTION

In bitcoin, the transaction is performed among peers in a trustless distributed environment by PoW algorithms that consume a huge amount of electrical energy [197]. The total consumption of energy in bitcoins is very high as reported if bitcoin were a country the energy consumption will be higher as the energy consumption in Iraq, Hong Kong, etc. [198] by the International Energy Agency. Bitcoin not only consumes a huge amount of energy but it also contributes to an extreme carbon footprint even bitcoin alone takes a big role in global warming up to 2 degrees in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

less than three decades. Thus, the solution to reduce energy consumption might be to redesign the infrastructure of blockchain and energy-efficient algorithms. Bitcoin energy consumption according to index [199] is shown in Table XIV.

F. SECURITY RISKS OF BLOCKCHAIN

Blockchain is a structure of blocks holding a comprehensive list of transaction histories open to everyone in a distributed P2P system. Therefore, there is always a risk that corrupt peers could manipulate transaction history data. The blockchain risks are divided into nine categories as shown in Table XV. The causes of these risks under the range of blockchain technology are also described in the given table. Among these nine categories, the first five come under the category of common risks of blockchain while the last four come under specific risks of blockchain.

VII. BLOCKCHAIN AND ITS INTEGRATION WITH EDGE COMPUTING

Edge computing, an open architecture that is enabling innovations for IoT, 5G, AI, etc., has been regarded as a solution for mitigating several security threats. Edge's distributed architecture safeguards connected systems from edge server to device by creating an additional layer of system security in which computation, control, storage, networking, and communications execute close to the services and the data sources. With edge, security resides directly in the local context, rather than a remote function.

Edge nodes protect cloud-based IoT and edge-based services by performing a wide range of security functions on a large number of interconnected devices even the smallest and the most resource-constrained. The security functions include providing a trusted distributed platform and execution environment for different services, managing and updating security credentials, scanning for malware, and timely distributing software patches quickly and at scale. Edge ensures trustworthy communication by detecting, validating, and reporting attacks. It can monitor the security status of nearby devices to quickly detect and isolate threats.

TABLE XIV. BITCOIN ENERGY CONSUMPTION

S. No.	Consumption	Total
01	Energy consumption per year	51.920 Terawatt-hour (TWh)
02	Global mining cost annually	$2,595,834,583
03	Worlds energy consumption Percentage	0.23%
04	Per transaction carbon footprint	274.29 kg of Carbon dioxide

If a security breach is detected, the edge provides trusted architecture components locally that enable real-time event response directly. The local context of the attack detection and response minimizes the disruption of services. Through edge computing network's scalability, modularity, capacity, and resource distribution, blockchain deployments for low-cost endpoints are very challenging. However, blockchain can be a potential solution for many security and other challenges in the edge computing. Therefore, blockchain convergence can complement edge computing with reliable and secure communication. The blockchain has also been considered one of the potential solutions to address many edge computing and IoT technical challenges.

One of the recent methods [55] for implementation of blockchains into IoT edge consists of cloud-blockchain hybrid architecture, in which a maximum volume of the IoT data is transported over the conventional IoT cloud-edge architecture. At the application level, the blockchain is employed where public accountability is required. The perception is to influence the low-latency data exchange of the conventional cloud and edge architecture, along with the immutable data storage functionalities of blockchains. As a result, the authors of [55] proposed a hybrid cloud-blockchain architecture shown in Figure 10, which would decrease the need of storing all the generated events in the blockchain. The architecture shown in this figure takes advantage of the accountability characteristics of the blockchain; however, it does not enforce service level agreements in a distributed manner for security all over the IoT edge.

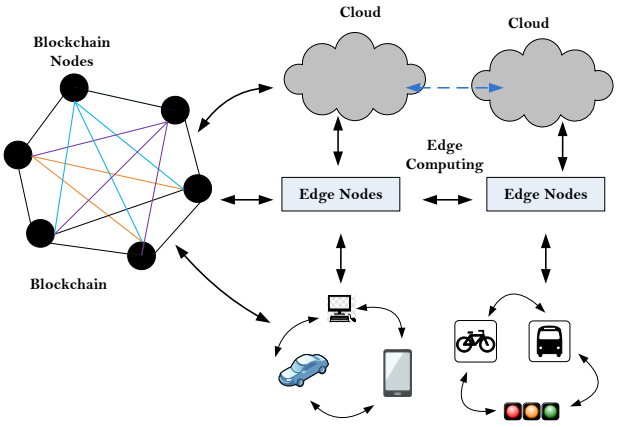

FIGURE 10. Cloud-blockchain hybrid architecture for edge

For harvesting the advantages of a blockchain-based edge computing, the horizontal scaling of blockchains over the edge is the utmost solution. Multiple locally deployed blockchains for multiple IoT edge networks, industrial segments, and heterogeneous networks are some other promising network architectures. The horizontal distribution of the blockchains and the blockchain-based solutions are vital to implementing blockchains at edges. The objective is to develop multiple blockchains or different blockchain modules, maintain communication records between IoT edges, and enforce service level agreements throughout the IoT edge network. Currently,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

different research developments are going on to implement shards of blockchains on the protocol level [58]. However, the existing architecture for a blockchain-based IoT edge is a hierarchical, multi-layered blockchain design, whereby permissioned and permissionless interaction between different blockchains can anticipatively aim to combine different segments of the IoT edge without overpowering other networks.

The layered-edge blockchain distributions are permissioned blockchains, where highly complex consensus algorithms are not required. Any external node can access the blockchain network after getting acceptance from any existing user. PBFT and other consensus protocols use voting methods that create higher network overhead than public blockchain consensus protocols such as PoW. Thus, private edge-tier blockchain architecture necessitates limitations on participating nodes and should be distributed in a localized network. Moreover, it is important to ensure the integrity of the data design considerations of the edge-tier blockchains. It becomes necessary to validate the content originality, as private-permissioned blockchains are unable to provide a similar

TABLE XV. NOMENCLATURE OF BLOCKCHAIN TECHNOLOGY THREATS

S. No.	Risk		Cause	Influence range
01	Common risks to blockchain	51% vulnerability	Consensus mechanism	Blockchain1.0
02		Private key security	Public-key encryption scheme	
03		Criminal activity	Cryptocurrency application	
04		Double spending	Transaction verification mechanism	
05		Transaction privacy leakage	Transaction design flaw	
06	Specific risks to blockchain	Criminal smart contracts	Smart contract application	Blockchain 2.0
07		Vulnerabilities in smart contract	Program design flaw	
08		Under-optimized smart contract	Program writing flaw	
09		Under-priced operations	EVM design flaw	

degree of distribution and immutability as open permissionless blockchains. Therefore, for the integrity of the network, it is necessary to make core-tier blockchains to keep hash of some recently generated blocks periodically in each edge-tier or utilize an independent public blockchain as an archive for hashes of all edge-tier networks. Some specific edge nodes in the edge-tier architecture act as gateways to make an interface between edge-tier and core-tier blockchains. Figure 11 represents the multi-tiered blockchain-based IoT edge architecture.

Communications among blockchains are being investigated at the protocol level [200]; though, blockchain applications created on platforms namely Ethereum and Hyperledger can communicate at the application level. Consequently, horizontally scaling architecture of blockchain is a feasible objective for giving distributed security to various IoT edge verticals.

An alternative methodology for keeping up transaction records is to store singular transactions in TDAGs. A case of such transaction graphs, or TDAGs, is found in the IOTA tangle [34]. TDAGs, including the IOTA tangle, incorporate Merkle trees of prior transaction IDs inside every data block. Approval time for newly generated transactions is hypothetically very small as arriving transactions just require to be approved by neighboring nodes of a transaction generating node. IOTA tangle can be viewed as a potential solution for the scalability in the IoT edge network, since tangle TDAG is not restricted to linear processing of transactions and keeps getting wider with higher volumes of incoming transactions. All arrival transactions are connected to various preceding transactions at the tip of the tangle, and every transaction is validated with preceding ones, accordingly decreasing the latency in public blockchain consensus algorithms. Apart from those architectures shown in Figures 10 and 11, an integrated blockchain and edge computing network architecture (cf. Figure 4) has been presented in the case study subsection (II-B). The blockchain-enabled edge architecture can solve the problems of naming and addressing, as well as certain security issues in edge computing. It also significantly simplifies the key management and distribution by assigning GUID and other communication protocols.

In addition, the blockchain is also important to reduce the dependency on platform providers. Next, we present some indispensable key features of blockchain that can solve many challenges and issues for edge computing including security issues.

A. ADDRESSING FOR THE DEVICES

Blockchain has a 20-byte address space while IPV6 has a 16-byte address space. The address generated by blockchain is 160-bit using the Elliptical Curve Digital Signature Algorithm (ECDSA). By using a 20-byte address, the blockchain can generate and assign addresses to about

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3037108, IEEE Access

1.46×10^{48} devices having an extremely low address collision likelihood. As a result, it is secure to allocate a global unique identity (GUI) which requires no signing in or individual authentication for edge devices. Blockchain eliminates the central authority and control of such as internet assigned numbers authority (IANA). At the moment, IANA is responsible for the allocation of IPv5 and IPv6 addresses. Edge devices with limited memory, power, and data processing resources will not be able to run the IPv6 stake.

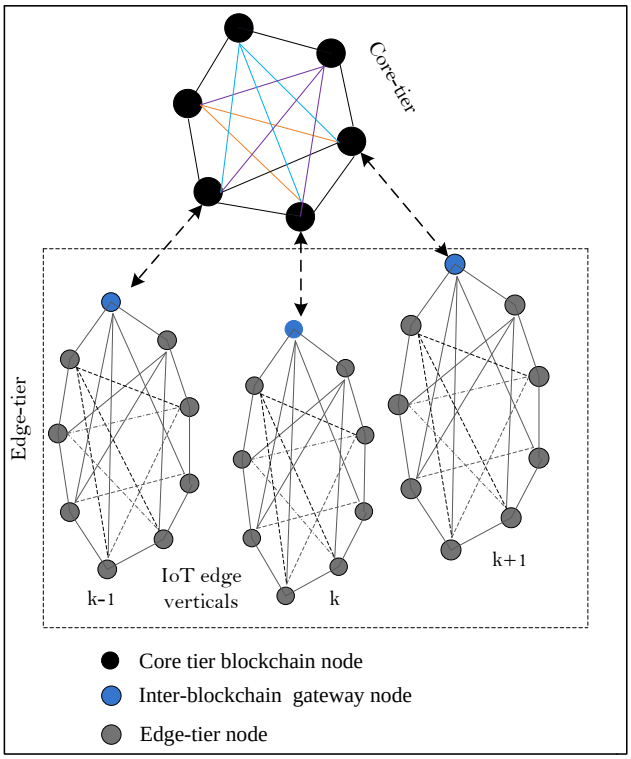

FIGURE 11. Multi-tiered blockchain-based IoT edge architecture

B. RELIABILITY

Blockchain is a decentralized, pooled, and immutable database of the records or ledger of every single transaction that has ever been executed and communicated between the different parties across a P2P network. It comprises data blocks chained together with each dependent hash value verified and timestamped by consensus of a majority of miners actively involved in authenticating and verifying transactions. Once the transactions are timestamped and verified by miners, they can never be erased or tampered (i.e. immutability). Besides, blockchain empowers device data tractability and liability. Reliability is the crucial characteristic of blockchain technology to edge computing.

C. IDENTITY AND ACCESS MANAGEMENT

Blockchain identity and access management (IAM) [201] for edge computing can address several issues efficiently, securely, and in a trustworthy way. With blockchain ownership and identity relationships of a device that keeps changing from a manufacturer to a consumer can be managed by assigning a global identifier for each device. The user ownership of a device is changed when the device is resold or compromised. Management of device attributes and associations is another challenge. All these challenges can be solved proficiently, effortlessly, and securely by using blockchain technology. At every point throughout the life cycle and supply chain of the mobile or edge device, the blockchain can offer trustworthy distributed management, control, and tracking.

D. VALIDATION, AUTHORIZATION, AND CONFIDENTIALITY

A smart contract can deliver distributed authentication logic and methods to offer single and multi-user authentication to an end-user. The smart contract based authorization access process is very simple in comparison with protocols e.g. OpenID, OAuth 2.0, open mobile alliance (OMA-DM), and lightweight machine-to-machine (LWM2M) for validation, authorization, and managing of edge devices. Smart contracts can be used to guarantee data privacy by setting access rules and time. The smart contract also controls the right to update, reconfigure, generate new key pairs, and change ownerships, as well as upgrade and patch the hardware/software components.

E. SECURE SOFTWARE DISTRIBUTION

Utilizing the secure-immutable storage characteristics of blockchain, codes can be installed in the devices securely and safely. Manufacturers can trace the system status and provide authenticated and immutable updates to the system. This functionality can be utilized to update the edge devices securely by edge computing.

F. AUTHENTICATION AND INTEGRITY OF DATA

The data transmitted by blockchain-enabled edge devices will always be verified and signed by the legitimate user holding a distinctive global unique identity (GUID), therefore confirming the authentication [202] and integrity of the transferred information. Besides, distributed ledgers are used to store all the transactions and can be securely traced in the blockchain.

G. SELF-GOVERNANCE

Blockchain technology enables cutting edge features of next-generation applications, by making conceivable the advancement of autonomous systems as a service. Blockchain gives devices the capability to interact and share data with no need for any central coordinator or management. The edge computing device can take advantage of such functionality to deliver device-skeptic and decoupled applications.

H. SECURE COMMUNICATIONS

Internet protocols such as MQTT, CoAp, XMPP, and routing protocols of RPL and 6LoWPAN are inherently vulnerable by architecture. We need to wrap these protocols

by other security algorithms or protocols like DTLS or TLS to direct messages and application protocols to achieve safe communication. Similar to the case with the routing protocols, IPsec is employed to give security to RPL and LowPAN schemes. These protocols are heavy and complex for the edge devices due to their high computational and memory needs.

Protocols such as MQTT, XMPP, and RPL have complicated central key management and distribution processes due to the use of public key infrastructure (PKI) protocol. Blockchain eliminates the key management and distribution by allocating GUID to each device and also simplifies other security protocols significantly such as DTLS, eradicating the need for key performance indicator (KPI) certificate exchange at the handshake phase. Henceforth, edge devices require the light-weight protocols that will run on the constrained computational and memory resources.

VIII. CONCLUSION

We have presented a comprehensive survey of edge computing and blockchain technologies, specifically, including the decentralized security model considerations and requirements for upgrading the applicability of the edge computing, and how various state-of-the-art technologies such as blockchain, AI, IoT, and ML can help for the construction of a secure edge computing paradigm. By the blockchain technology, smart contracts, consensus mechanisms, AI and ML algorithms, it can be anticipated that QoS and security of the edge computing can be significantly upgraded in the prospective future. Furthermore, we have presented some possible potential blockchain solutions to main security attacks in edge computing. In addition to the existing solutions shown in Figures 10 and 11, the potential solution reported in this review paper is the proposed integrated blockchain and edge computing architecture (cf. Figure 4), where smart contracts are used for edge device registration, data storage, service, and resource management along with authentication of transaction data. Finally, we summarized and identified some open research issues and challenges for offering reliable, proficient, and scalable security services of edge computing. Thus, further investigation is still essential to build an edge-blockchain framework for massive collaboration, reconfiguration, energy efficiency, scalability, and flexibility. Surely, it is inevitable for the need of decreasing the imparity amongst edge computing, blockchain architectures, and diverse protocols to achieve the ultimate objective of secure and reliable edge computing networking services.

REFERENCES

[1] C. Bloom, J. Tan, J. Ramjohn, and L. Bauer, "Self-driving cars and data collection: Privacy perceptions of networked autonomous vehicles," in Proc. USENIX Conf. Usable Privacy and Security (SOUPS), Santa Clara, USA, Jul. 2017, pp. 357–375.

[2] M. Caprolu, R. D. Pietro, F. Lombardi, and S. Raponi, "Edge computing perspectives: Architectures, technologies, and open security issues," in Proc. IEEE Int. Conf. Edge Compt. (EDGE), Milan, Italy, Jul. 2019, pp. 116-123.

[3] R. Roman, J. Lopez, and M. Mambo, "Mobile edge computing, fog et al.: A survey and analysis of security threats and challenges," Future Gen. Comput. Syst., vol. 78, no. 2, pp. 680-698, Jan. 2018.

[4] Z. Xiong, Y. Zhang, D. Niyato, P. Wang, and Z. Han, "When mobile blockchain meets edge computing," IEEE Commum. Magazine, vol. 56, no. 8, pp. 33-39, Apr. 2018.

[5] R. Yang, F. R. Yu, P. Si, Z. Yang, and Y. Zhang, "Integrated blockchain and edge computing systems: A survey, some research issues, and challenges," IEEE Commun. Surveys & Tutor., vol. 21, no. 2, pp. 1508-1532, Jan. 2019.

[6] S. Khan, S. Parkinson, and Y. Qin, "Fog computing security: A review of current applications and security solutions," J. of cloud Comput. vol. 6, no. 1, pp. 19, Dec. 2017.

[7] D. Puthal, et al. "Fog computing security challenges and future directions [energy and security]," IEEE Consumer Electronics Magazine, vol. 8, no. 3, pp. 92-96, Apr. 2019.

[8] M. Mukherjee et al., "Security and privacy in fog computing: Challenges," IEEE Access, vol. 5, pp. 19293-19304, Sep. 2017.

[9] P. Zhang, M. Zhou, and G. Fortino, "Security and trust issues in fog computing: A survey," Future Gen. Computer Syst., vol. 88, pp. 16-27, Nov. 2018,

[10] A. Shahzad, and M. Hussain, "Security issues and challenges of mobile cloud computing," Int. J. of Grid and Distributed Comput., vol. 6, no. 6, pp. 37-50, Dec. 2013.

[11] S. N. Shirazi, A. Gouglidis, A. Farshad, and D. Hutchison, "The extended cloud: Review and analysis of mobile edge computing and fog from a security and resilience perspective," IEEE J. on Selected Areas in Commun., vol. 35, no. 11, pp. 2586-2595, Oct. 2017.

[12] R.-H. Hsu, J. Lee, T. Q. S. Quek, and J. C. Chen, "Reconfigurable security: Edge-computing-based framework for IoT," IEEE Netw., vol. 32, no. 5, pp. 92-109, Sep. 2018.

[13] M. B. Mollah, M. A. Azad, and A. Vasilakos, "Security and privacy challenges in mobile cloud computing: Survey and way ahead," J. of Netw. and Computer Apps., vol. 84, pp. 38-54, Apr. 2017.

[14] A. N. Khan, M. M. Kiah, S. U. Khan, and S. A. Madani, "Towards secure mobile cloud computing: A survey," Future Gen. Computer Syst., vol. 29, no. 5, pp. 1278-1299, Jul. 2013.

[15] H. Suo, Z. Liu, J. Wan, and K. Zhou, "Security and privacy in mobile cloud computing," in IEEE Int. Wireless Commun. and Mobile Comput. Conf. (IWCMC), Sardinia, Italy, Jul. 2013, pp. 655-659.

[16] M. Ichaba, "Security threats and solutions in mobile ad hoc networks; A review," Universal J. of Communs. and Netw., vo. 6, no. 2, pp. 7-17, 2018.

[17] N. Islam, Z. A. Shaikh, "Security issues in mobile ad hoc network," Wireless Netw. and Security, Berlin, Heidelberg, pp. 49-80, 2013.

[18] I. Yaqoob, E. Ahmed, A. Gani, S. Mokhtar, M. Imran, and S. Guizani, "Mobile ad hoc cloud: A survey," Wir. Commun. Mob. Comput., vol. 16, pp. 2572– 2589, Jul. 2016.

[19] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," IEEE Internet of Things J., vol. 3, no. 5, pp. 637-646, Oct. 2016.

[20] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computing—A key technology towards 5G," ETSI white paper, vol. 11, no. 11, pp. 1-6, Sep. 2015.

[21] B. Varghese, N. Wang, S. Barbhuiya, P. Kilpatrick, and D. S. Nikolopoulos, "Challenges and opportunities in edge computing," in IEEE Int. Conf. on Smart Cloud (SmartCloud), NY, USA, Nov. 2016, pp. 20-26.

[22] S. Jeong, O. Simeone, and J. Kang, "Mobile edge computing via a UAV-mounted cloudlet: Optimization of bit allocation and path planning," IEEE Trans. Vehicular Techn., vol. 67, no. 3, pp. 2049-2063, Mar. 2018.

[23] S. Dustdar, C. Avasalcai, and I. Murturi, "Edge and fog computing: Vision and research challenges," in IEEE Int. Conf. on Service-oriented Syst. Eng. (SOSE), San Francisco, USA, Apr. 2019, pp. 9696-9609.

[24] M. Alrowaily and Z. Lu, "Secure edge computing in IoT systems: Review and case studies," in Proc. IEEE/ACM Symp. on Edge Comput. (SEC), Seattle, WA, USA, Oct. 2018, pp. 440-444.

[25] W. Yu, F. Liang, X. He, W. G. Hatcher, C. Lu, J. Lin, and X. Yang, "A survey on the edge computing for the Internet of Things," IEEE Access, vol. 6, pp. 6900-9619, Nov. 2017.

[26] M. Yahuza et al., "Systematic review on security and privacy requirements in edge computing: State of the art and future research opportunities," IEEE Access, vol. 8, pp. 76541-76567, 2020.

[27] I. Eyal, A. E. Gencer, E. G. Sirer, and R. V. Renesse, "Bitcoin-NG: A scalable blockchain protocol," in Proc. Usenix Conf. on Netw. Syst. Design and Implementation (NSDI), USA, Feb. 2016, pp. 45–59.

[28] M. Pilkington, "Blockchain technology: Principles and applications," Research Handbook on Digital Transformations, edited by F. Xavier Olleros and M. Zhegu. Edward Elgar, Montreal, Canada: Sep. 2016.

[29] D. Drescher, *Blockchain **B**asics*. Berkeley, CA, USA: vol. 276, Apress, 2017.

[30] C. Cachin, "Architecture of the hyperledger blockchain fabric," in Proc. Workshop on Distributed Cryptocurrencies and Consensus Ledgers, vol. 310, no. 4, Jul. 2016, pp. 1- 4.

[31] D. D. Maesa, and P. Mori, "Blockchain 3.0 applications survey," J. of Parallel and Distributed Comput., vol. 138, pp. 99-114, Apr. 2020.

[32] M. Nofer, P. Gomber, O. Hinz, and D. Schiereck, "Blockchain," Business & Inf. Syst. Eng., vol. 59, no. 3, pp. 183-187, Feb. 2017.

[33] M. Conoscenti, A. Vetro, and De Martin, "Blockchain for the internet of things: A systematic literature review," in IEEE/ACS Int.Conf. of Computer Syst. and Apps. (AICCSA), Agadir, Morocco, Nov. 2016, pp. 1-6.

[34] S. Popov, "The tangle," vol. 4, pp. 1-28, no. 3. https://assets.ctfassets.net/r1dr6vzfxhev/2t4uxvsIqk0EUau6g2sw0g/45eae33637ca92f85dd9f4a3a218e1ec/iota1_4_3.pdf

[35] V. Buterin, "Ethereum white paper: A next-generation smart contract & decentralized application platform," vol. 3, no. 37, Jan. 2014.

[36] L. M. Goodman, "Tezos: A self-amending crypto-ledger position paper," White Paper, vol. 3, pp. 1-17, Aug. 2014.

[37] M. Samaniego, U. Jamsrandorj, and R. Deter, "Blockchain as a service for IoT," in IEEE Int. Conf. on internet of things (iThings) and IEEE green comput. and communs. (GreenCom), and IEEE cyber, physical and social comput. (CPSCom), and IEEE smart data (SmartData), Chengdu, China, Dec. 2016, vol. 15, pp. 433-436.

[38] K. Biswas and V. Muthukkumarasamy, "Securing smart cities using blockchain technology," in IEEE Int. Conf. on high-performance comput. and communs., IEEE smart city, IEEE data science and systems (HPCC/SmartCity/DSS), Sydney, NSW, Australia, Dec. 2016, pp. 1392-1393.

[39] Al-Jaroodi and N. Mohamed, "Blockchain in industries: A survey," IEEE Access, vol. 7, pp. 36500–36515, 2017.

[40] J. A. Jaoude and R. G. Saade, "Blockchain applications—Usage in different domains," IEEE Access, vol. 7, pp. 45360–45381, 2019.

[41] M. Holbl, M. Kompara, A. Kamisalic, and N. Zlatolas, "A systematic review of the use of blockchain in healthcare," Symmetry, vol. 10, no. 10, pp. 470-492, Oct. 2018.

[42] Z. Zheng, X. Shaoan, H.-N. Dai, X. Chen, and H. Wang, "Blockchain challenges and opportunities: A survey," Inter. J. of Web and Grid Services, vol. 14, no. 4, pp. 352-375, Oct. 2018.

[43] G. Zyskind, and O. Nathan, "Decentralizing privacy: Using blockchain to protect personal data," in IEEE Security and Privacy Workshop, San Jose, CA, USA, May 2015, pp. 180-184.

[44] I.-C. Lin, T.-C. Liao, "A survey of blockchain security issues and challenges," Int. J. Netw. Security. vol. 19, no. 5, pp. 653-660, Sep. 2017.

[45] X. Li, P. Jiang, T. Chen, X. Luo, and Q. Wen, "A survey on the security of blockchain systems," Future Gen. Computer Syst., vol. 107, pp. 841-53, Jun. 2020.

[46] Q. Feng, D. He, S. Zeadally, M. K. Khan, and N. Kumar, "A survey on privacy protection in the blockchain system," J. Netw. Comput. Appl., vol. 126, pp. 45-58, 2019.

[47] J. Park, "Blockchain security in cloud computing: Use cases, challenges, and solutions," Symmetry, vol. 9, no. 8, pp. 1-13, 2017.

[48] Y. Xiao, N. Zhang, W. Lou, and Y. T. Hou, "A survey of distributed consensus protocols for blockchain Networks," in *IEEE* Commun. Surveys & Tutors., vol. 22, no. 2, pp. 1432-1465, Jan. 2020.

[49] L. Ismail and H. Materwala, "A review of blockchain architecture and consensus protocols: Use cases, challenges, and solutions," Symmetry, vol. 11, no. 10, pp. 1-47, Sep. 2019.

[50] S. Sankar, M. Sindhu, and M. Sethumadhavan, "Survey of consensus protocols on blockchain applications," in Proc. IEEE Int. Conf. on Advanced Comput. and Commun. Syst. (ICACCS), Coimbatore, India, Jan. 2017, pp. 1-5.

[51] M. S. Ali et al., "Applications of blockchains in the internet of things: A comprehensive survey," IEEE Commun. Surveys & Tutor., vol. 21, no. 2, pp. 1676-1717, Dec. 2018.

[52] A. Ouaddah, A. A. Elkalam, and A. A. Ouahman, "Fairaccess: A new blockchain-based access control framework for the internet of things," Security Commun. Netw., vol. 9, no. 18, pp. 5943–5964, 2016.

[53] A. Bahga and V. K. Madisetti, "Blockchain platform for the industrial internet of things," J. Software Eng. and Apps., vol. 9, no. 10, pp. 533-546, Sept./Oct. 2016.

[54] M. A. Khan and K. Salah, "IoT security: Review, blockchain solutions, and open challenges," Future Gen.Computer Syst., vol. 82, pp. 395-411, May 2018.

[55] G. Sagirlar et al., "Hybrid-iot: Hybrid blockchain architecture for internet of things-Pow sub-blockchains," in IEEE Int. Conf. Internet of Things (iThings), and IEEE Green Comput. and Communs. (GreenCom), and IEEE Cyber, Physical, and Social Comput. (CPSCom), and IEEE Smart Data (SmartData), Halifax, NS, Canada, Jul. 2018, pp. 1007-1016.

[56] A. Stanciu, "Blockchain-based distributed control system for edge computing," in Int. Conf. on Control Syst. and Computer Sci. (CSCS), Bucharest, Romania, May. 2017, pp. 667-671.

[57] M. Conoscenti, A. Vetro, De Martin, and J. Carlos, "Blockchain for the internet of things: A systematic literature review," in Proc. of the IEEE/ACS Int. Conf. of Computer Syst. and Apps. (AICCSA), Agadir, Morocco and Piscataway, USA, Nov/Dec. 2016, pp. 1–6.

[58] M. S. Ali, M. Vecchio, and F. Antonelli, "Enabling a blockchain-based IoT edge," IEEE Internet of Things Magazine, vol. 1, no. 2, pp. 24-29, Dec. 2018.

[59] A. Dorri et al., "LSB: A lightweight scalable blockchain for IoT security and privacy," J. of Parallel and Distributed Comput., vol. 134, pp. 180-197, Dec. 2019.

[60] P. Mendki, "Blockchain enabled IoT edge computing: Addressing privacy, security and other challenges," in Proc. Int. Conf. Blockchain Technology (ICBCT'20), Assoc. for Comput. Machinery, New York, NY, USA, 2020, pp. 63–67.

[61] P. Bhattacharya, S. Tanwar, R. Shah, and A. Ladha, "Mobile edge computing-enabled blockchain framework—a survey," in Proc. ICRIC, Springer, Cham, vol. 597, pp. 797-809, Nov. 2019.

[62] E. Elrom, "NEO blockchain and smart contracts," The Blockchain Developer, Apress, Berkeley, CA, USA, Jul. 2019, pp. 257-298.

[63] G. Bu, O. Gurcan, and M. Potop-Butucaru, "G-IOTA: Fair and confidence aware tangle," in IEEE INFOCOM, IEEE Conf. Computer Commun. Workshop, Paris, France, Apr. 2019, pp. 644-649.

[64] A. Yousefpour et al., "All one needs to know about fog computing and related edge computing paradigms: A complete survey," J. of Syst. Architecture, vol. 98, pp. 289-330, Sep. 2019.

[65] "Fog Computing and the internet of things: Extend the cloud to where the things are," Cisco, pp. 1-5, 2016.

[66] J. Acharya and S. Gaur, "Edge compression of GPS data for mobile IoT," in Proc. IEEE Fog World Congress (FWC), Santa Clara, USA

May 2018, pp. 1-6.

[67] F. Adams, "Open-fog reference architecture for fog computing," Openfog Consortium and Architecture Working, Feb. 2017, pp. 1-162.

[68] S.-C. Hung, H. Hsu, S.-Y. Lien, and K.-C. Chen, "Architecture harmonization between cloud radio access networks and fog networks," IEEE Access, vol. 3, pp. 3019-3034, Dec. 2015.

[69] IBM News Releases, "IBM and Nokia Siemens networks announce world first mobile edge computing platform," Barcelona, Spain, Feb. 2013. (http://www-03.ibm.com/press/us/ en/pressrelease/40490.wss)

[70] S. Singh, Y.-C. Chiu, Y. Tsai and J.-S. Yang, "Mobile edge fog computing in 5G era: Architecture and implementation," in Proc. IEEE Inter. Comput. Symp. (ICS), Chiayi, Taiwan, Feb. 2017, pp. 731-735.

[71] P. Mach and Z. Becvar, "Mobile edge computing: A Survey on architecture and computation offloading," IEEE Commun. Surveys & Tutor., vol. 19, no. 3, pp. 1628-1656, Mar. 2017.

[72] M. T. Beck, M. Werner, S. Feld, and T. Schimpe, "Mobile edge computing: A taxonomy," in Proc. Int. Conf. on Adv. in Future Internet (AFIN), Lisbon, Portugal, Nov. 2014, pp. 48-55.

[73] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "Mobile edge computing: Survey and research outlook," IEEE Commun. Surveys & Tutor., pp. 1-31, Jan. 2017.

[74] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computing - A key technology towards 5G," ETSI white paper, vol. 11, no. 11, pp. 1-16, Sep. 2015.

[75] A. Khan, W. Kellerer, K. Kozu, and M. Yabusaki, "Network sharing in the next mobile network: TCO reduction, management flexibility, and operational independence," IEEE Commun. Mag., vol. 49, no. 10, pp. 134-142, Oct. 2011.

[76] M. Hoffmann and M. Staufer, "Network virtualization for future mobile networks: General architecture and applications," in Proc. IEEE Inter. Conf. Commun. Workshops (ICC), Kyoto, Japan, Jul. 2011, pp. 1-5.

[77] M. Yang et al., "Software-defined and virtualized future mobile and wireless networks: A survey," Mobile Netw. Appl. vol. 20, pp. 4–18, Sep. 2015.

[78] S. Barbarossa, S. Sardellitti, and P. Di Lorenzo, "Communicating while computing: Distributed mobile cloud computing over 5G heterogeneous networks," IEEE Sig. Proc. Mag., vol. 31, no. 6, pp. 45-55, Nov. 2014.

[79] "Smart cells revolutionize service delivery," Intel White Paper, vol. 2, pp. 1-7, May. 2013. (www.intel.com/go/commsinfrastructure).

[80] G. Simmons, G. A. Armstrong, and M. G. Durkin, "An exploration of small business website optimization: Enablers, influencers, and an assessment approach," Inter. Small Business Jour. vol. 29, no. 5, 534–561, Feb. 2011.

[81] M. A. Hoque, M. Siekkinen, and J. K. Nurminen, "Energy-efficient multimedia streaming to mobile devices-A survey," IEEE Commun. Surveys & Tutor., vol. 16, no. 1, pp. 579-597, Nov. 2014.

[82] R. Sharma, S. Kumar, and M. C. Trivedi, "Mobile cloud computing: A needed shift from cloud to the mobile cloud," in Proc. Inter. Conf. and Comput. Intel. and Commun. Netws, Mathura, India, Sep. 2013, pp. 536-539.

[83] S. S. Qureshi, T. Ahmad, K. Rafique, and Shuja-ul-Islam, "Mobile cloud computing as a future for mobile applications - implementation methods and challenging issues," in Proc. IEEE Inter. Conf. Cloud Comput. and Intel. Syts, Beijing, China, Sep. 2011, pp. 467-471.

[84] P. Zhang and Z. Yan, "A QoS-aware system for mobile cloud computing," in Proc. IEEE Inter. Conf. Cloud Comput. and Intel. Syst. Beijing, Sep. 2011, pp. 518-522.

[85] S. Khan, E. Ahmad, M. Shiraz, A. Gani, W. A. Wahab, and M. A. Bagiwa, "Forensic challenges in mobile cloud computing," in Proc. Inter. Conf. Compt., Commun. and Control Techn. (I4CT), Langkawi, Sep. 2014, pp. 343-347.

[86] S. Hakak, S. A. Latif, and G. Amin, "A review on mobile cloud computing and issues in it," Inter. J. of Computer Apps., vol. 75, no. 11, pp. 1-4, Jan. 2013.

[87] M. Shiraz and A. Gani, "A lightweight active service migration framework for computational offloading in mobile cloud computing," J. Supercomput. vol. 68, no. 2, pp. 978–995, May 2014.

[88] H. T. Dinh, C. Lee, D. Niyato and P. Wang, "A survey of mobile cloud computing: Architecture, applications, and approaches," Wirel. Commun. Mob. Comput., vol. 13, pp. 1587-1611, Oct. 2011.

[89] D. M. Shila, W. Shen, Y. Cheng, X. Tian, and X. S. Shen, "AMCloud: Toward a secure autonomic mobile ad hoc cloud computing the system," IEEE Wir. Commun. vol. 24, no. 2, pp. 74-81, Apr. 2017.

[90] V. Balasubramanian and A. Karmouch, "An infrastructure as a service for mobile ad-hoc cloud," in Proc. IEEE Annual Compt. and Commun. Workshop and Conf. (CCWC), Las Vegas, Jan. 2017, pp. 1-7.

[91] M. Jang, M. S. Park, and S. C. Shah, "A mobile ad hoc cloud for an automated video surveillance system," in Proc. Inter. Conf. Comput., Netw. And Commun. (ICNC), Santa Clara, Jan. 2017, pp. 1001-1005.

[92] S. C. Shah, "Mobile Ad Hoc computational grid: Opportunities and challenges," in Proc. IEEE Military Comms. Conf., San Diego, Nov. 2013, pp. 848-857.

[93] Z. Pang, L. Sun, Z. Wang, E. Tian, and S. Yang, "A survey of cloudlet based mobile computing," in Proc. Int. Conf. Cloud Compt. and Big Data (CCBD), Shanghai, Nov. 2015, pp. 268-275.

[94] A. S. Jaiswal, V. Thakare, and S. Sherekar, "Performance-based analysis of cloudlet architectures in mobile cloud computing," Inter. J. of Comput. App., vol. 1, pp. 4-10, Apr. 2015.

[95] H. Guo and J. Liu, "Collaborative computation offloading for multiaccess edge computing over fiber–wireless networks," IEEE Trans. Vehicular Techn., vol. 67, no. 5, pp. 4514-4526, May 2018.

[96] Y. Jararweh, L. Tawalbeh, F. Ababneh, A. Khreishah, and F. Dosari, "Scalable cloudlet-based mobile computing model," Procedia Computer Sci., (MobiSPC), vol. 34, Aug. 2014, pp. 434-441.

[97] X. Sun and N. Ansari, "Green cloudlet network: A sustainable platform for mobile cloud computing," IEEE Trans. Cloud Compt., vol. 8, no. 1, pp. 180-192, Jan. 2020.

[98] Q. Fan and N. Ansari, "On cost-aware cloudlet placement for mobile edge computing," IEEE/CAA J. of Automatica Sinica, vol. 6, no. 4, pp. 926-937, July 2019.

[99] T. Lynn, P. Rosati, A. Lejeune, and V. Emeakaroha, "A preliminary review of enterprise serverless cloud computing (Function-as-a-Service) platforms," in Proc. IEEE Inter. Conf. Cloud Compt. Techn. and Sci. (CloudCom), Hong Kong, Dec. 2017, pp. 162-169.

[100] R. Rajan, and A. Paul, "Serverless architecture-A revolution in cloud computing," in Proc. Inter. Conf. Advan. Compt. (ICOAC), Chennai, India, Dec. 2018, pp. 88-93.

[101] S. S. Gill et al., "Transformative effects of IoT, blockchain and artificial intelligence on cloud computing: Evolution, vision, trends and open challenges," IEEE Internet of Things J., vol. 8, pp. 100-118, Dec. 2019.

[102] E. M. Dogo et al., "Taking cloud computing to the extreme edge: A review of mist computing for smart cities and industry 4.0 in Africa," Edge comput., Springer, Cham, pp. 107-132, Nov. 2019.

[103] M. Liyanage, C. Chang, and S. N. Srirama, "mePaaS: Mobile-embedded platform as a service for distributing fog computing to edge nodes," in Proc. Inter. Conf. Parallel and Distrib. Comput., Apps. and Techn. (PDCAT), Guangzhou, Jun. 2016, pp. 73-80.

[104] J. S. Preden, K. Tammemae, A. Jantsch, M. Leier, A. Riid, and E. Calis, "The benefits of self-awareness and attention in fog and mist computing," IEEE Compt., vol. 48, no. 7, pp. 37-45, Jul. 2015.

[105] J. S. Preden, "Evolution of mist computing from fog and cloud computing," THIINECT White Paper, pp. 1-4, Aug. 2016. (http://www.thinnect.com/static/2016/08/cloud-fog-mist-computing-062216.pdf).

[106] A. A. Abdellatif, A. Mohamed, C. F. Chiasserini, M. Tlili, and A. Erbad, "Edge computing for smart health: Context-aware

approaches, opportunities, and challenges," IEEE Network, vol. 33, no. 3, pp. 196-203, Mar. 2019.

[107] E. Deelman, K. Vahi, G. Juve, M. Rynge, S. Callaghan, P. J. Maechling, R. Mayani, W. Chen, R. F. da Silva, M. Livny, and K. Wenger, "Pegasus, a workflow management system for science automation," Future Gener. Comput. Syst., vol. 46, pp. 17-35, May 2015.

[108] K. Wolstencroft et al., "The Taverna workflow suite: Designing and executing workflows of web services on the desktop, web, or in the cloud," Nucleic Acids Res., vol. 41, no. 1, pp. 557-561, Jul. 2013.

[109] S. Kartakis and J. A. McCann, "Real-time edge analytics for cyber-physical systems using compression rates," in Proc. Int. Conf. Auto. Comput. (ICAC), Philadelphia, USA, Jul. 2014, pp. 153–159.

[110] I. Santos, M. Tilly, B. Chandramouli, and J. Goldstein, "DiAl: Distributed streaming analytics anywhere, anytime," in Proc. VLDB Endow., vol. 6, pp. 1386-1389, Aug. 2013.

[111] L. Xu, Z. Wang, and W. Chen, "The study and evaluation of ARM-based mobile virtualization," Inter. J. of Distrib. Sensor Netw., vol. 11, no. 7, pp. 308-310. Jul. 2015.

[112] J. Andrus, C. Dall, A.V. Hof, O. Laadan, and J. Nieh, "Cells: A virtual mobile smartphone architecture," in Proc. Symp. Operating Syst. Principles, New york, USA, Oct. 2011, pp. 173-187.

[113] B. S. Rad, H. J. Bhatti, and M. Ahmadi, "An Introduction to docker and analysis of its performance," Int. J. of Comput. Sci. and Ntw. Security, vol. 17, no. 3, pp. 228-235, Mar. 2017.

[114] C. Pahl, A. Brogi, J. Soldani, and P. Jamshidi, "Cloud container technologies: A state-of-the-art review," IEEE Trans. Cloud Comput., vol. 7, no. 3, pp. 677-692, Sept. 2019.

[115] B. Varghese, L. T. Subba, L. Thai, and A. Barker, "DocLite: A docker-based lightweight cloud benchmarking tool," in Proc. IEEE/ACM Int. Symp. on Cluster, Cloud, and Grid Comput., Cartagena, Jul. 2016, pp. 213-222.

[116] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," IEEE Internet of Things J., vol. 3, no. 5, pp. 637-646, Oct. 2016.

[117] L. Zhang et al., "Named data networking (ndn) project," Relatorio Tecnico NDN-0001, Xerox Palo Alto Research Center-PARC, vol. 157, pp. 158, Oct. 2010

[118] D. Raychauduri, K. Nagaraja, and A. Venkataramani, "MobilityFirst: A robust and trustworthy mobility-centric architecture for the future internet," SIGMOBILE Mob. Comput. Commun. Rev., vol. 16, no. 3, pp. 2–13, Jul. 2012.

[119] B. T. Smith et al., "Features and functioning of data abstraction assistant, a software application for data abstraction during systematic reviews," Research Synthesis Methods, vol. 10, no. 1, pp. 2-14, Mar. 2019.

[120] L. Cardelli and P. Wegner, "On understanding types, data abstraction, and polymorphism," ACM Comput. Surv. vol. 17, no. 4, pp. 471–523, Dec. 1985.

[121] B. Varghese, N. Wang, S. Barbhuiya, P. Kilpatrick and D. S. Nikolopoulos, "Challenges and opportunities in edge computing," in IEEE Int. Conf. on Smart Cloud (SmartCloud), New York, Dec. 2016, pp. 20-26.

[122] C.-H. Hong and B. Varghese, "Resource management in fog/edge computing: A survey on architectures, infrastructure, and algorithms," ACM Compt. Surv., vol. 52, no. 5, pp. 1-37, Oct. 2019.

[123] J. Feld, "PROFINET - Scalable factory communication for all applications," in Proc. IEEE Int. Workshop on Factory Comm. Systs, Proc., Vienna, pp. 33-38, Sep. 2004.

[124] F. daCosta and B. Henderson, "Rethinking the internet of things: A scalable approach to connecting everything," Nature Springer, USA, pp. 157-158, Oct. 2013.

[125] A. Ahmed and E. Ahmed, "A survey on mobile edge computing", Int. Conf. Intelligent Systems and Control (ISCO), Coimbatore, 2016, pp. 1-8.

[126] G. Wang et al., "Interactive medical image segmentation using deep learning with image-specific fine tuning," IEEE Trans. Med. Imaging, vol. 37, no. 7, pp. 1562-1573, Jul. 2018.

[127] K. R. Jackson et al., "Performance analysis of high-performance computing applications on the Amazon web services cloud," in Proc. IEEE Inter. Conf. on Cloud Comput. Techn. and Sci., Indianapolis, USA, pp. 159-168, Feb. 2011.

[128] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, "The cost of a cloud: Research problems in data center networks," SIGCOMM Comput. Commun. Rev. 39, pp. 68–73, Jan. 2009.

[129] A. P. Miettinen and J. K. Nurminen, "The energy efficiency of mobile clients in cloud computing," in Proc. USENIX conf. hot topics in cloud Comput. USENIX Association, USA, vol. 4, no. 4, pp. 1-7, 2010.

[130] N. Ding, D. Wagner, X. Chen, A. Pathak, Y. C. Hu, and A. Rice, "Characterizing and modeling the impact of wireless signal strength on smartphone battery drain," SIGMETRICS Perform. Eval. Rev., vol. 41, no. 1, pp. 29–40, Jun. 2013.

[131] B. Varghese, L. T. Subba, L. Thai, and A. Barker, "DocLite: A docker-based lightweight cloud benchmarking tool," in Proc. IEEE/ACM Int. Symp. on Cluster, Cloud, and Grid Comput., Cartagena, Jul. 2016, pp. 213-222.

[132] B. Varghese, O. Akgun, I. Miguel, L. Thai, and A. Barker, "Cloud benchmarking for performance," in Proc. IEEE Inter. Conf. on Cloud Compt. Techn. and Sci., Feb. 2015, pp. 535-540.

[133] F. Brian, Cooper, A. Silberstein, E. Tam, R. Ramakrishnan, and R. Sears, "Benchmarking cloud serving systems with YCSB," in Proc. ACM Symp. on Cloud comput., Assoc. for Computing Machinery, New York, NY, USA, Jun. 2010, pp. 143–154.

[134] D. Sabella, A. Alleman, and E. Liao, etc., "Edge Computing: From standard to actual infrastructure deployment and software development," ETSI White paper, pp. 1-41, Oct. 2019.

[135] V. Q. Rodriguez, F. Guillemin, and A. Boubendir, "Automating the deployment of 5G network slices using ONAP," in Proc. Inter. Conf. NoF, Rome, Italy, Feb. 2019, pp. 32-39.

[136] P. Simoens, L. Van Herzeele, F. Vandeputte, and L. Vermoesen, "Challenges for orchestration and instance selection of composite services in distributed edge clouds," in Proc. IFIP/IEEE Inter. Symp. on Intgr. Netw. Manag. (IM), Ottawa, Jul. 2015, pp. 1196-1201.

[137] J. Byrne et al., "A review of cloud computing simulation platforms and related environments," in Proc. Int. Conf. Cloud Comput. and Services Sci., Porto, Portugal, Apr. 2017, vol. 2, pp. 679-691.

[138] W. Tian et al., "Open-source simulators for cloud computing: Comparative study and challenging issues," Simul. Model. Prac. and Theory, vol. 58, pp. 239-254, Nov. 2015.

[139] A. Ahmed, A. S. Sabyasachi, "Cloud computing simulators: A detailed survey and future direction", in IEEE int. advance comput. Conf. (IACC), Gurgaon, India, Feb. 2014, pp. 866-872.

[140] M. Ashouri, F. Lorig, P. Davidsson, and R. Spalazzese, "Edge computing simulators for IoT system design: An analysis of qualities and metrics," Future Internet, vol. 11, pp. 235, Nov. 2019.

[141] S. Svorobej et al., "Simulating fog and edge computing scenarios: An overview and research challenges," Future Internet, vol. 11, no. 3, pp. 55, Feb. 2019.

[142] "Multi-access edge computing (MEC): Framework and reference architecture," ETSI Group, vol. 2, no. 1, pp. 1-21, Jan. 2019. (http://www.etsi.org/standards-search).

[143] "Multi-access Edge Computing (MEC)," ETSI Group, vol. 2, no. 1, pp. 1-21, Nov. 2019. https://www.etsi.org/technologies/multi-access-edge-computing).

[144] E. M. Dogo et al., "Taking cloud computing to the extreme edge: A review of mist computing for smart cities and industry 4.0 in Africa," Edge comput., Springer, Cham, pp. 107-132, Nov. 2019.

[145] J. Chen and X. Ran, "Deep learning with edge computing: A review," Proc. of the IEEE, vol. 107, no. 8, pp. 1655-1674, Aug. 2019.

[146] O. B. Sezer, E. Dogdu, and A. M. Ozbayoglu, "Context-aware computing, learning, and big data in the internet of things: A survey," IEEE Internet of Things J., vol. 5, no. 1, pp. 1-27, Feb. 2018.

[147] J. Park et al., "Wireless network intelligence at the edge," Proc. of the IEEE, vol. 107, no. 11, pp. 2204–2239, Sep. 2019.

[148] Q. Mao, F. Hu, and Q. Hao, "Deep learning for intelligent wireless networks: A comprehensive survey," IEEE Commun. Surveys & Tutor., vol. 20, no. 4, pp. 2595-2621, Oct. 2018.

[149] O. Simeone, "A very brief introduction to machine learning with applications to communication systems," IEEE Trans. Cognitive Comms. and Netw., vol. 4, no. 4, pp. 648-664, Dec. 2018.

[150] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," in Proc. of the IEEE, vol. 107, no. 8, pp. 1738-1762, Aug. 2019.

[151] M. S. Mahdavinejad, M. Rezvan, M. Barekatain, P. Adibi, P. Barnaghi, and A. P. Sheth, "Machine learning for the internet of things data analysis: A survey," Digital Commun., and Netw., vol. 4, no. 3, pp. 161-175, Aug. 2018.

[152] W. Guo and J. S. Wang, "Deep multimodal representation "Deep multimodal representation learning: A Survey," IEEE Access, vol. 7, pp. 63373-63394, May 2019.

[153] P.-B. Diego, and G.-B. Bertha, "A survey of methods for distributed machine learning," Prog. Artificial Inteligence, vol. 2, no. 1, pp. 1–11, Nov. 2012.

[154] J. Dean et al., "Large scale distributed deep networks," Adv. in Neural Informatics, Proc. Syst., pp. 1223-1231, Dec. 2012.

[155] J. K. Aggarwal and Lu Xia, "Human activity recognition from 3D data: A review," Pattern Recognition Letters, vol. 48, pp. 70-80, Oct. 2014.

[156] H. F. Nweke, Y. W. Teh, M. A. Al-garadi, and U. Z. Alo, "Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges," Expert Syts. with Apps., vol. 105, pp. 233-261, Sep. 2018.

[157] A. Mikołajczyk and M. Grochowski, "Data augmentation for improving deep learning in an image classification problem," in Proc. Inter. Interdisciplinary Ph.D. Workshop (IIPhDW), Swinoujscie, May 2018, pp. 117-122.

[158] G. Wang et al., "Interactive medical image segmentation using deep learning with image-specific fine tuning," IEEE Trans. Med. Imaging, vol. 37, no. 7, pp. 1562-1573, Jul. 2018.

[159] S. S. Gill et al., "Transformative effects of IoT, blockchain and artificial intelligence on cloud computing: Evolution, vision, trends and open challenges," IEEE Internet of Things J., vol. 8, pp. 100-118, Dec. 2019.

[160] M. Alrowaily and Z. Lu, "Secure edge computing in IoT systems: Review and case studies," in Proc. IEEE/ACM Symp. on Edge Comput. (SEC), Seattle, WA, USA, Oct. 2018, pp. 440-444.

[161] Y. Xiao, Y. Jia, C. Liu, X. Cheng, J. Yu, and W. Lv, "Edge computing security: State of the art and challenges," Proc. of the IEEE, Aug. 2019, vol. 107, no. 8, pp. 1608-1631.

[162] Y. Shi, S. Abhilash, and K. Hwang, "Cloudlet mesh for securing mobile clouds from intrusions and network attacks," in Proc. IEEE Int. Conf. Mobile Cloud Comput. Services and Engi., San Francisco, CA, Mar. 2015, pp. 109-118.

[163] R. K. C. Chang, "Defending against flooding-based distributed denial-of-service attacks: A tutorial," IEEE Coms. Mag., vol. 40, no. 10, pp. 42-51, Oct. 2002.

[164] R. Mohammadi, R. Javidan, and M. Conti, "SLICOTS: An SDN-based lightweight countermeasure for TCP SYN flooding attacks," IEEE Trans. Netw. and Service Mang., vol. 14, no. 2, pp. 487-497, June 2017.

[165] G. Gursun, M. Sensoy, and M. Kandemir, "On context-aware DDoS attacks using deep generative networks," in Proc. Int. Conf. on Comput. Commun. and Netw. (ICCCN), Hangzhou, Oct. 2018, pp. 1-9.

[166] Y. You, M. Zulkernine, and A. Haque, "Detecting flooding-based DDoS attacks," in Proc. IEEE Int. Conf. on Communs. Glasgow, Aug. 2007, pp. 1229-1234.

[167] Y.-H. Hu, H. Choi, and H.-A. Choi, "Packet filtering to defend flooding-based DDoS attacks [Internet denial-of-service attacks]," in Proc. IEEE/Sarnoff Symp. on Advances in Wired and Wireless Commun., Princeton, USA, Apr. 2004, pp. 39-42.

[168] L. Feinstein, D. Schnackenberg, R. Balupari, and D. Kindred, "Statistical approaches to DDoS attack detection and response," in Proc. DARPA Inf. Survivability Conf. and Expos., Washington, DC, USA, Apr. 2003, vol. 1, pp. 303-314.

[169] M. Nooribakhsh and M. Mollamotalebi, "A review on statistical approaches for anomaly detection in DDoS attacks," Inf. Security J.: A Global Perspective, vol. 29, no. 3, pp. 118-133, Mar. 2020.

[170] X. Yuan, C. Li, and X. Li, "DeepDefense: Identifying DDoS attack via deep learning," in Proc. IEEE Int. Conf. on SMARTCOMP, Hong Kong, Jun. 2017, pp. 1-8.

[171] C.-J. Hsieh and T.-Y. Chan, "Detection DDoS attacks based on neural-network using Apache Spark," in Proc. Int. Conf. on Applied Syst. Innovation (ICASI), Okinawa, Aug. 2016, pp. 1-4.

[172] S. Vinson, R. Stonehirsch, J. Coffman, and J. Stevens, "Preventing zero-day exploits of memory vulnerabilities with guard lines," in Proc. Workshop on SSPREW9, Assoc. for Comput. Machinery, New York, USA, Dec. 2019, no. 2, pp. 1–11.

[173] V. Sharma et al., "A consensus framework for reliability and mitigation of zero-day attacks in IoT," Security and Commun. Netw., vol. 2017, pp. 1-24, Jan. 2017.

[174] Y. Xiao et al., "Edge computing security: State of the art and challenges," Proc. of the IEEE, vol. 107, no. 8, pp. 1608-1631, June 2019.

[175] A. Diro and N. Chilamkurti, "Distributed attack detection scheme using deep learning approach for the internet of things," Fut. Gen. Comput. Syst., vol. 82, pp. 761-768, May 2018.

[176] V. Vassilakis et al., "Security analysis of mobile edge computing in virtualized small cell networks," in Proc. IFIP Int. Conf. AI Apps. and Innov. Springer, Thessaloniki, Greece, Sep. 2016, pp. 653-665.

[177] N. Kube and D. Drescher, "Blockchain basics: A non-technical introduction in 25 steps," Finance Mark Portf Manag, vol. 32, pp. 329–331, 2018.

[178] S.-R. Nexhibe, E. Ramadani, F. Idrizi, V. Misimi, "Blockchain: General overview of the architecture, security and reliability," J. of Natural Sci. and Math. UT, vol. 3, no. 5, pp. 64-68, Nov. 2018.

[179] J. Al-Jaroodi and N. Mohamed, "Industrial applications of blockchain," in IEEE Annual Comput. and Commun.Workshop and Conf. (CCWC), Las Vegas, USA, Jan. 2019, pp. 0550-0555.

[180] Z. Zheng, X. Shaoan, H.-N. Dai, X. Chen, and H. Wang, "Blockchain challenges and opportunities: A survey," Inter. J. of Web and Grid Services, vol. 14, no. 4, pp. 352-375, Oct. 2018.

[181] T. Hanke, M. Movahedi, and D. Williams, "DFINITY," Techn. Overview Series, Consensus System, May 2018.

[182] S. Lee, D. Kim, D. Kim, S. Son, and Y. Kim, "Who spent my EOS? on the (in)security of resource management of EOS.IO," in Proc. the USENIX Conf. on Offens. Techn. (WOOT). Santa, Clara, USA, Aug. 2019, pp. 1-11.

[183] J. Poon and T. Dryja, "The bitcoin lightning network: Scalable off-chain instant payments," Lightining Network, vol. 9, no. 2, pp. 1-59, Jan. 2016.

[184] B. Singhal, G. Dhameja, and P. S. Panda, *Beginning Blockchain: A Beginner's Guide to Building Blockchain Solutions*. Berkeley, CA, USA: Apress, Jul. 2018.

[185] J. Sousa, A. Bessani, and M. Vukolic, "A byzantine fault-tolerant ordering service for the hyperledger fabric blockchain platform," in IEEE/IFIP int. conf. on dependable syst. and netw. (DSN), Luxembourg Jun. 2018, pp. 51-58.

[186] E. Androulaki et al., "Hyperledger fabric: A distributed operating system for permissioned blockchains," in Proc. EuroSys Conference, New York, USA, Apr. 2018, pp. 1-15.

[187] H. Jin, X. Dai, and J. Xiao, "Towards a novel architecture for enabling interoperability amongst multiple blockchains," in Proc. IEEE Int. Conf. on Distributed Comput. Systems (ICDCS),Vienna, Austria, Jul. 2018, pp. 1203-1211.

[188] E. B. Hamida, K. L. Brousmiche, H. Levard, and E. Thea, "Blockchain for enterprise: Overview, opportunities and challenges,"

HAL Archives, vol. 10, no. 4, pp. 1-7, Jul. 2017.

[189] T. C. Yan, P. Schulte, and D. L. K. Chuen, "Blockchain—From the public to private," In Handbook of Blockchain, Digital Finance, and Inclusion; Elsevier, Amsterdam, Netherlands, vol. 2, pp. 145–177, Aug. 2018.

[190] G. Greenspan, "MultiChain Private Blockchain—White Paper," pp. 1-17, Jul. 2015. (URl: http://www. multichain. com/download/MultiChain-White-Paper. pdf).

[191] G. Wood, "Ethereum: A secure decentralized generalized transaction ledger," Ethereum Project Yellow Paper, vol. 151, pp. 1-32, Apr. 2014.

[192] W. Fang et al., "Digital signature scheme for information non-repudiation in the blockchain: A state of the art review," J. Wireless Commun. Netw., vol. 56, pp. 1-5, Dec. 2020.

[193] M. Macdonald, L. Thorrold, and R. Julien, "The blockchain: A comparison of platforms and their uses beyond bitcoin," COMS4507-Adv. Compt. and Netw. Sec., vol. 26, pp. 1-17, May, 2017.

[194] B. W Nyamtiga, J. C. Sicato, S. Rathore, Y. Sung, and J. H. Park, "Blockchain-Based Secure Storage Management with Edge Computing for IoT," Electronics, vol. 8, no. 8, pp. 828-850, Aug. 2019.

[195] D. Puthal, R. Ranjan, A. Nanda, P. Nanda, P. Jayaraman, and A. Y. Zomaya, "Secure authentication and load balancing of distributed edge data centers," J. of Parallel and Distr. Compt., vol.124, pp. 60-69, Oct. 2019.

[196] A. Lei et al., "Blockchain-based dynamic key management for heterogeneous intelligent transportation systems," IEEE Internet of Things J., vol. 4, no. 6, pp. 1832-1843, Aug. 2017.

[197] K. J. O'Dwyer and D. Malone, "Bitcoin mining and its energy footprint," in Proc. IET Irish Signals & Systems Conf. and Int. Conf. Inf. and Commun. Techn. (ISSC/CIICT), Limerick, Jun. 2014, pp. 280-285.

[198] S. Sanju, S. Sankaran, and K. Achuthan, "Energy comparison of blockchain platforms for the internet of things," in Proc. IEEE Int. Symp. on Smart Electronic Systems (ISES), Hyderabad, India, Dec. 2018, pp. 235-238.

[199] A. deVries, "Bitcoin's growing energy problem," Joule, vol. *2*, no. 5, pp. 801-805, May 2018.

[200] L. Kan et al., "A Multiple Blockchains Architecture on Inter-Blockchain Communication," IEEE Int. Conf. Software Quality, Reliability and Security Companion, Lisbon, Portugal, 2018. pp. 139–45.

[201] G. Zyskind, O. Nathan, and A. Pentland, "Decentralizing privacy: using blockchain to protect personal data," in Proc. IEEE Sec. and PriV. Workshop, San Jose, CA, Jul. 2015, pp. 180-184.

[202] A. Chandak and N. K. Ray, "A review of the load balancing in fog computing," in IEEE Conf. on Information Technology (ICIT), Bhubaneswar, India, Dec. 2019, pp. 460-465.

SHOWKAT AHMAD BHAT received his bachelor degree in Electronics and Communication from Punjab Technical University, Punjab, India and his master degree from Lovely Professional University, Punjab, India. He joined Wireless Communication and Signal Processing (WCSP) Laboratory, National Tsing Hua University to pursue his PhD degree under the supervision of Prof. Chong-Yung Chi in fall 2019. His primary research interests include Edge Computing, Blockchain Technology, and Wireless Sensor Networks.

ISHFAQ BASHIR SOFI received a Master's degree in electronics from Kashmir University, Jammu, and Kashmir, India, in 2016. He is currently pursuing an M.Tech degree in electronics and communication engineering from Lovely Professional University, Punjab, India. He is currently involved in research work on the energy efficiency of massive MIMO communication, Edge computing, and Blockchain Technologies. He is also a student member of IEEE community.

CHONG-YUNG CHI (Fellow, IEEE) received the BS degree from the Tatung Institute of Technology, Taipei, Taiwan, in 1975, the master's degree from National Taiwan University, Taipei, Taiwan, in 1977, and the Ph.D. degree from the University of Southern California, Los Angeles, California, in 1983 all in electrical engineering. Currently, he is a professor of National Tsing Hua University, Hsinchu, Taiwan. He has published more than 140 peer-reviewed conference papers, three book chapters, and two books, journal papers (mostly in the IEEE Trans. Signal Processing), more than including a recent textbook, Convex Optimization for Signal Processing 240 technical papers, including more than 85 and Communications from Fundamentals to Applications, CRC Press, 2017 (which has been popularly used in a series of invited intensive short courses at the top-ranking universities in Mainland China since 2010 before its publication). Recently, he received 2018 IEEE Signal Processing Society Best Paper Award. His current research interests include signal processing for wireless communications, convex analysis and optimization for blind source separation, biomedical and hyperspectral image analysis, and graph signal processing. He has been a Technical Program Committee member for many IEEE sponsored and co-sponsored workshops, symposiums and conferences on signal processing and wireless communications, including co-organizer and general co-chairman of 2001 IEEE Workshop on Signal Processing Advances in Wireless Communications (SPAWC). He was an associate editor (AE) for four IEEE Journals, including the IEEE Trans. Signal Processing for nine years (5/2001–4/2006, 1/2012–12/2015), and he was a member of Signal Processing Theory and Methods Technical Committee (SPTM-TC) (2005–2010), a member of Signal Processing for Communications and Networking Technical Committee (SPCOM-TC) (2011–2016), and a member of Sensor Array and Multichannel Technical Committee (SAM-TC) (2013–2018), IEEE Signal Processing Society.