Arabian Journal for Science and Engineering (2022) 47:9801–9831
https://doi.org/10.1007/s13369-021-06348-2

RESEARCH ARTICLE-COMPUTER ENGINEERING AND COMPUTER SCIENCE

AI-Based Mobile Edge Computing for IoT: Applications, Challenges, and Future Scope

Ashish Singh[1] · Suresh Chandra Satapathy[1] · Arnab Roy[1] · Adnan Gutub[2]

Received: 11 August 2021 / Accepted: 28 October 2021 / Published online: 3 January 2022

Abstract

New technology is needed to meet the latency and bandwidth issues present in cloud computing architecture specially to support the currency of 5G networks. Accordingly, mobile edge computing (MEC) came into picture as novel emerging solutions to overcome many cloud computing issues. In this contemporary technology, the computation server and processing units are nearby edge servers to reduce latency, increase the network bandwidth and reduce energy consumption in user devices. These features can integrate with several domains such as the internet of things, artificial intelligence (AI), federated learning (FL) and fog computing, etc., to make the system more robust, elastic, efficient, and accurate. Regardless of the advantages, MEC faces several challenges, including security and privacy, deployment protocols, and offloading management. Although, various studies have been found tuning MEC to solve such challenges, the literature provide more ideas for smart developments toward applications particularly using FL and AI. Most researches miss combining interesting aspects of MEC, such as machine learning and deep learning approaches limiting works to only single aspect. Thus, a literature work is needed to focus on all the aspects of MEC together. This study aims to present a comprehensive survey on MEC by providing all necessary information, including network architecture, advantages, objectives, access technologies, deployment templates, characteristics, and many more. The work is not limited to only MEC background but also covers the AI and FL approaches used within MEC, allowing mobile phones to learn a shared predictive model collaboratively. This survey also provides information regarding security and privacy challenges as well as attacks on MEC and their solutions. The applications of MEC illustrate different sectors where MEC is applicable further highlighting open issues and challenges to be investigated.

Keywords Mobile edge computing (MEC) · Internet of things (IoT) · Mobile communications · Edge technology · AI and FL technology · Security and privacy challenges

1 Introduction

Mobile terminals such as smartphones, tablets, and computers have become an integral part of our lives. Increasing the number of smartphones is directly proportional to the increase of network traffic. The exponential growth of devices and data traffic creates issues in terms of high latency, high bandwidth, and lack of data storage capabilities. The smart technological evolutions in mobile, laptops, and tablets give rise to the highly demanding applications and services based on mobile technology. With the shortage of timing, most of the users required processed data or outcomes in a very short period. The high demanding applications provide real-time services by the processing of real-time data. The size of the real-time data is huge because, over time, new data is generated accumulatively. Thus, handling huge amount of real-time data by mobile phones is very difficult in short time period and limited capability, i.e., serving applications containing high processing tasks, as also increase battery consumption restricting users from enjoying demanding applications. Until now, mobile cellular

✉ Adnan Gutub
aagutub@uqu.edu.sa

Ashish Singh
ashishashish307@gmail.com

Suresh Chandra Satapathy
suresh.satapathyfcs@kiit.ac.in

Arnab Roy
arnab.roy100@gmail.com

[1] School of Computer Engineering, KIIT Deemed To Be University, Bhubaneswar 751024, Odisha, India

[2] Computer Engineering Department, Umm Al-Qura University, Makkah, Saudi Arabia

has limited capabilities, including memory, bandwidth, and infrastructure. These constrain increase energy consumption and service latency issues. The problem present in the current systems may be assisted by including the concept of emerging technologies such as Edge networks [1], EdgeIoT [2], and mobile edge computing. In this work, the problem present in the current system is addressed thoroughly by including the concept of mobile and edge computing (MEC).

Mobile Cloud Computing (MCC) integrates cloud computing concepts with the mobile environment [3]. MCC provides many capabilities to the mobile devices in which a user can access computing and storage resources from a powerful centralized cloud through the Internet [4]. MCC brings advantages such as increased battery life by offloading extensive computation applications, using sophisticated applications, and providing higher data storage capabilities [5]. However, MCC also faced several challenges such as high latency, low coverage, lag in data transmission, and security vulnerability. These challenges made the system inconvenient and less suitable needing novel concept or architecture to be adopted, especially handling real-time cases providing fast responses enjoying high Quality of Service (QoS) [6].

In 2009, the concept of edge computing was introduced [7] with main aim to address MCC's challenges. This concept brings computational devices near to users similar to WiFi hotspot scenarios, i.e., instead of Internet connection, the approach/setup provides computational other services. MEC offers MCC capabilities by deploying cloud resources such as storage and processing capacity on the edge server or edge of any network. The basic idea behind MEC is that all the applications and services are hosted near the cellular network to reduce transmission time and latency. This can support end-users in accessing swift and powerful computing resources, flexible and rapid deployment of new applications, energy efficiency environment, high storage capacity, high mobility, location, and context awareness applications.

In the MEC environment, several computing nodes (node servers) are deployed in a distributed manner. Any user can connect with the MEC nodes according to the proximity of the two devices. The node server takes care of the heavy computational tasks that users submit and sends back to them. This is faster than cloud services, because computing nodes are closer to the user devices. European Telecommunications Standards Institute (ETSI) is the organization that provides standard MEC network architecture and definition, supported by mobile network operators such as Docomo, Vodafone, IBM and manufacturer's such as Nokia and Huawei [8].

MEC plays a significant role in supporting high communication, better computing capabilities, controlled information sharing, and better content delivery in 5G networks. The use of small mobile base stations and wireless access points deployed with computational capacity makes ubiquitous mobile computing [9] environment. The MEC provides several low cost and efficient solutions for the Internet of Things (IoT) [10]. These IoT solutions contain interrelated internet-connected objects capable of sharing information over wireless media without human intervention [11]. The MEC solution brings the cloud services close to the IoT device [12].

Additionally, artificial intelligence (AI) in deep learning networks is a powerful tool and technique that addresses the problems and empowers real-time resource management [13–16] for efficient IoT-MEC environment [17–19]. The inclusion of AI techniques in MEC improve the quality and accuracy of the system to take decisions faster than humans. Higher-latency and lower-throughput problems present in traditional machine learning systems are needed to make a new robust machine learning model. The AI-based Federated Learning (collaborative learning) [20] works in distributed manner in which machine learning algorithms are performed over the edge node without sharing the data [21], different than counting-based secret-sharing [22]. This AI-MEC approach has overcome many algorithmic and technical challenges present in the machine learning model. FL deployment in IoT-enabled technology gives more robust and fault-tolerant frameworks. This emerging machine learning approaches provide better privacy solutions and progressive application deployment templates. The key characteristics of this efficient approach are all the information to be locally processed [23], as the training data reside on the local device. The FL approach separates the learning models' needs from the necessity to store the data. The main advantages of AI-based FL include high data security/privacy, lower service latency, data diversity, hardware efficiency, real-time data analysis, and many more. The FL-based MEC approach gives solution for deploying real-time online gaming, ultra-high definition video stream, and Virtual Reality (VR) applications [24]. Regardless of AI-based federated learning advantages in several fields, it has some limitations. In this approach, the data is distributed on multiple servers increasing the attack surface. Various devices are integrated to build a model in federated learning such that device-specific characteristics may reduce the performance, i.e., of the federated learning model. Orchestrator can be counted as another challenge in federated learning approach too [25].

Recent advancement such as Network Function Virtualization (NFV) [26–28], Information-Centric Cloud (ICN) [29–31], and Software Defined Network (SDN) in the field of computer networks also helped in the deployment of efficient MEC environments. These discussions demonstrated that the AI-based FL approach for the MEC-IoT environment is an important research area that needs more attention. In the literature, some of the survey papers are identified, which helps develop such systems comprehensively, unlike specific open-service confidentiality reviews such as twitter privacy studies [32]. In other words, a systematic review paper covering all the aspects of this AI-MEC area is not identified appropriately. In this regard, some of the important research work (survey papers) are listed out in Table 1. In this table, three symbols are used. "–" symbol signifies less discussion of the topic. The "✓" and " × " symbols denote that the particular topic is covered and the topic is not covered, respectively. This table (Table 1) compared all the existing research work (survey papers) to help identify gaps and limitations.

This survey paper tried to overcome the existing limitations present in the previous related study papers. The structure of this survey paper is shown in Fig. 1, contributing the following summarized points, as will be discussed later.

- This survey paper reviewed and compared the existing study papers related to AI- and FL-based MEC for IoT highlighting their limitations.
- MEC background is discussed including deployment and technical developments, network architecture, and specific MEC advantages. This review paper also included FL-based AI approaches applicable to MEC for IoT environments.
- The security and privacy challenges with countermeasures are discussed. This section is based on the discussed background and AI/FL approaches, i.e., linked to MEC. This discussion also facilitated different attacks possible in AI-based MEC environment as well as development of many online/offline applications.
- Based on the complete literature survey and phenomena of AI-based MEC, some open issues and urgent upcoming challenges have been discussed as needed to be addressed in the future.

The paper's remaining sections are organized as follows. Section 2 discusses the existing solutions for AI- and FL-based MEC environment. The background of the MEC is deliberated in Sect. 3. Section 4 presents the different FL-based AI approaches for MEC. Security and privacy challenges of AI- and FL-based MEC, including countermeasures, are elaborated in Sect. 5. Several specific security attacks are listed out in Sect. 6. The applications of MEC appropriate to different sectors are discussed in Sect. 7. Open issues and challenges for the future are discussed in Sect. 8. The conclusion of the paper is presented in Sect. 9.

2 Existing Solutions for AI- and FL-Based MEC

The AI-based MEC environment provides an intelligent system in which mobile or IoT devices are communicated to provide efficient services. The AI-based distributed FL approach is considered as additional machine learning technology that will be helpful for development of security and privacy applications, i.e., serving learning models deployed in MEC for IoT systems. Several works are identified aiming to provide numerous solutions for security and privacy in AI- and FL-based MEC environment [47–55], as critically summarized in Table 2.

Unique potential privacy issues present in MEC wireless networks are discussed in paper [60]. They present usage pattern privacy problems and location privacy matters coping current technology involvement in public systems [86]. Therefore, author of [60] also proposed a scheduling algorithm that can effectively perform task offloading along with maintaining privacy. This procedure is proposed based on the Constrained Markov Decision Process (CMDP) framework that achieves low latency and efficient energy consumption, i.e., in devices that maintain their performance and privacy.

FedMec model in the MEC environment is proposed by Zhang et al. in paper [87], which allows maintaining the privacy of the training data and an efficient FL protocol. Physical-layer assisted privacy-preserving scheme [88] provides both efficiency and privacy. In this [88] structure, edge server is responsible for offloading the task and proactively sends out jamming signals to stop eavesdroppers from obtaining valuable information.

In [89], researchers show that combining blockchain in multi-domain networks enables secure topology in MEC, as secure collaboration is possible in multiple domains. In [90], authors illustrate the inference attack with the help of Wald's sequential hypothesis testing. This experiment concludes existence of privacy risk in the current MEC system. Similarly, an offloading scheme is proposed to deal with this issue, which also preserves the system's privacy and performs cost-effective offloading operations. Xu et al. [71] propose the BeCome method in IoT-based MEC, which uses Blockchain to reduce the time needed during offloading and provide efficient consumption of energy.

In [2] the author proposes an architecture called edgeIoT. This architecture effectively handles the huge amount of traffic generated by IoT devices. This would reduce the traffic load in the network and provide lower latency. The computational resource comparison of edgeIoT with the traditional IoT architecture shows better service provisioning. Relatively, the paper [91] shows proliferation of using AI/ML techniques which improve the edge computing paradigms, i.e., with greater efficiency of network bandwidth usage, reduced latency, and ultra-reliability for the future 6G networks. The dynamic environment demands high mobility and low latency. AI-driven Heterogeneous MEC architecture has been discussed in paper [92] to achieve such superior demands.

In [93] highlights, the critical role of AI ensuring network security in 5G and beyond raised the possibility of security

Table 1 Comparison of the survey with the existing survey papers

Paper	Year	Discussed topics	Existing Solutions	Background of MEC	FL based AI approaches	Security & privacy challenges	Counter measures	Security attacks	Application of MEC	Open issues
Abbas et al. [33]	2017	MEC Solutions, Application, Background, Infrastructure, Security and Privacy	✓	✓	×	✓	✓	✓	✓	✓
Mao et al. [34]	2017	Communication Model, Security and Privacy Challenges, Resource Management, Applications	–	–	×	✓	–	×	–	✓
Mach et al. [35]	2017	Communication Model, MEC Solutions, Scheme for Offloading Computational Resources, Applications	✓	×	×	×	×	×	✓	×
Allah et al. [36]	2017	MEC Solutions, Edge Location Trade off, Application, Server Locations	–	×	×	×	×	×	–	–
Taleb et al. [37]	2017	Background, Architecture, Security and privacy challenges, Network Orchestration	–	✓	×	–	–	×	✓	×
Porambage et al. [38]	2018	Existing solutions, Background, IoT, Integration Technologies	✓	✓	×	–	–	–	✓	✓
Moura et al. [39]	2018	Background, Game Theory, Application	–	✓	×	×	×	×	✓	✓
Yousefpour et al. [40]	2019	Existing Solutions, Background, Similar Paradigms, Framework Security and Privacy	✓	✓	×	✓	✓	✓	–	✓
Mehrabi et al. [41]	2019	Existing Solutions, Background, Computation offloading, Content Caching	–	✓	×	×	×	×	✓	✓
Li et al. [42]	2019	Existing Solutions, Background, System Components, Open Source System	✓	✓	✓	×	×	×	✓	×
Aledhari et al. [43]	2020	Architecture Optimizing, FL Model, Application	×	–	✓	×	×	×	✓	✓
Pham et al. [44]	2020	Background, Non Orthogonal Multiple Access Wireless Power Transfer, UAV Communication, Security and Privacy Challenges	–	✓	×	–	–	–	✓	–
Lim et al. [45]	2020	Background, FL, Security and Privacy Challenges, Resource Allocation	×	✓	✓	✓	–	–	–	✓
Spinelli et al. [46]	2020	Existing Solutions, Standardization, Flexible Provisioning, Industrial Approach	✓	×	×	×	×	×	✓	×
This survey	2021	Existing MEC Solution, Background of MEC, FL based AI approaches, Security and Privacy Challenges, Countermeasures, Security attacks, Applications, Open Issues	✓	✓	✓	✓	✓	✓	✓	✓

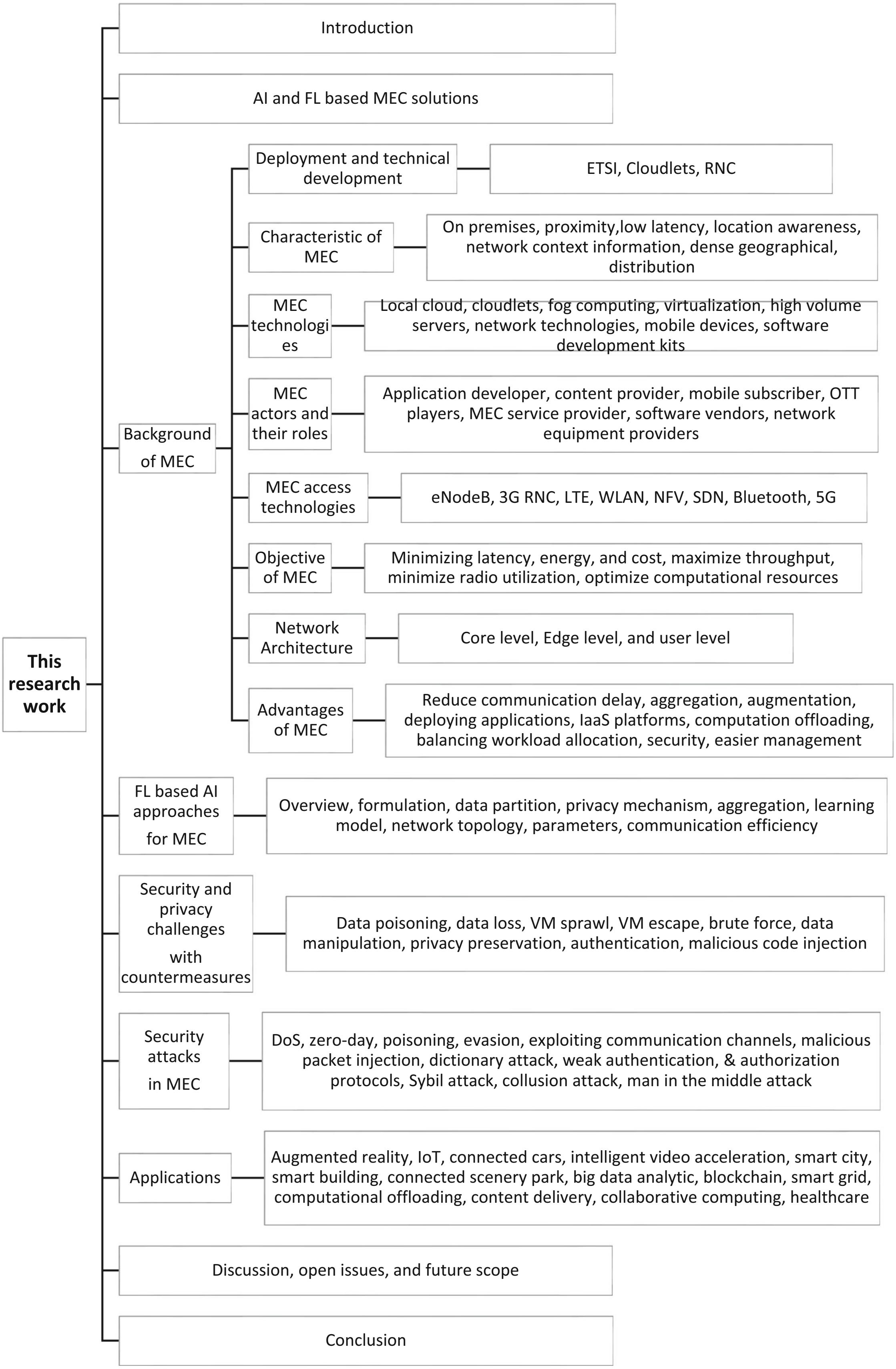
This research work
Introduction
AI and FL based MEC solutions
Background of MEC
Deployment and technical development
ETSI, Cloudlets, RNC
Characteristic of MEC
On premises, proximity,low latency, location awareness, network context information, dense geographical, distribution
MEC technologies
Local cloud, cloudlets, fog computing, virtualization, high volume servers, network technologies, mobile devices, software development kits
MEC actors and their roles
Application developer, content provider, mobile subscriber, OTT players, MEC service provider, software vendors, network equipment providers
MEC access technologies
eNodeB, 3G RNC, LTE, WLAN, NFV, SDN, Bluetooth, 5G
Objective of MEC
Minimizing latency, energy, and cost, maximize throughput, minimize radio utilization, optimize computational resources
Network Architecture
Core level, Edge level, and user level
Advantages of MEC
Reduce communication delay, aggregation, augmentation, deploying applications, IaaS platforms, computation offloading, balancing workload allocation, security, easier management
FL based AI approaches for MEC
Overview, formulation, data partition, privacy mechanism, aggregation, learning model, network topology, parameters, communication efficiency
Security and privacy challenges with countermeasures
Data poisoning, data loss, VM sprawl, VM escape, brute force, data manipulation, privacy preservation, authentication, malicious code injection
Security attacks in MEC
DoS, zero-day, poisoning, evasion, exploiting communication channels, malicious packet injection, dictionary attack, weak authentication, & authorization protocols, Sybil attack, collusion attack, man in the middle attack
Applications
Augmented reality, IoT, connected cars, intelligent video acceleration, smart city, smart building, connected scenery park, big data analytic, blockchain, smart grid, computational offloading, content delivery, collaborative computing, healthcare
Discussion, open issues, and future scope
Conclusion

Fig. 1 The structure of this survey work

risk associated with the AI benefits envisioned. Relatively, paper [94] provided comprehensive review of IoT, including IoT-based MEC architecture, enabling technologies, and security and privacy issues. Moreover, it also includes integrating fog/edge computing with IoT. Interestingly, work [72] focused on privacy-preserving problems in the smart grid network. The approach can lead to better data protection along with acceptable performance. It proposed the use of blockchain as system to join all the entities in the grid network. The approach also introduced a special type of Node (SNs), which validates the participating nodes. In research [73] the focused investigation was on offloading problems presenting strategy designed to create proper balance between effective use and privacy, which divides the process into two phases in a smart way. Furthermore, to manage the huge amount of finely grained complex sensing data, the authors [66] proposed using crowdsensing system. Similarly, the Internet of Vehicles (IoV) applications need to have higher bandwidth, lower latency, and higher reliability, and MEC that can meet the needs, i.e., of such applications. The authors of [74] proposed collaborative scheduling strategy to help allocate intended computing resources in case of IoT-based MEC. This approach dealt with offloading MEC problems when dealing with user tasks and better computational resource allocation in MEC.

Paper [75] studies the time consumption issue in MEC, and it also focuses on maintaining the system's privacy. The research proposes offloading method with better security and lower time consumption. Exploration [95] presented an intelligent Game-theoretic privacy-aware task allocation solution. This solution is applicable for the Social Sensing-based Edge Computing system, which optimized QoS to ensure that the privacy requirements of end-users are met. In study [96] model, a three-layer privacy protection architecture has been worked out as Edge Computing Architecture (ECA), at the edge of the network framework, i.e., based on ontology of system behavior to be highly dynamic. Differently, work [97] presented Privacy-aware Edge Computing for providing privacy in Social Sensing-based Edge Computing system which did not show much applicability.

Survey papers [33, 98–100] presented comprehensive overview and MEC research outlook. MEC deployment concerns, cache-enabled problems, MEC mobility management, green MEC, and security challenges have all been listed as potential research directions in addition to smart IoT works [101]. In the paper [76], the author dealt with the issue of resource allocation when deploying the MEC-intrusion detection system. Mathematical modeling is used in the proposed allocation mechanism. Likewise, the author [102] proposed security solution that uses reinforcement learning to deal with the privacy issues within the intended MEC system. The author presented caching collaboration scheme that can also perform lightweight authentication to deal with smart attacks when performing mobile offloading, theoretically comparable to semi-authentication of multimedia strategy [103]. Work [104] presented Honeypots which deal with harmful define-to-define communication. The responsibility of Honeypots is to detect, track and isolate malicious activity in the device-to-device network. Accordingly, paper [105] discussed security issues due to third-party MEC providers. Similarly, paper [67] proposed security architecture of Vehicular Ad-hoc Networks to ensure VANET data's authenticity utilizing combining blockchain and MEC. This VANET architecture used three layers, namely perception, edge computing, and service layer, with the perception responsibility ensuring data security during transmission through the blockchain. This work [67] compared the encryption-based security with the physical layer security trying to solve MEC-based IoT challenges via encryption physical layer security approaches. The solution includes secure wiretap coding, resource allocation, signal processing, secure key generation, authentication, and multi-node cooperation. To ensure maximum security and solve decision-making problems in fog and MEC, other paper [106] proposed use of hesitant fuzzy which added unpractical complexity.

Syamkumar et al. [68] considered the problem presented in geographic distributed MEC micro data centers. They proposed incremental deployment model which composed of Voronoi Cell-based analysis. They also discovered that tower deployment in rural areas is consistent when compared to urban areas. Likewise, Li et al. [69] proposed the middlebox approach to deal with the low latency issues during MEC deployment. Proxy ARP, GTP (GPRS Tunneling Protocol), Repackaging, Traffic Redirection via DNS, and Stateful Tracking of GTP Tunnel are some of the approaches used in the proposed model [69]. Martin et al. [107] projected a mathematical model for determining the deployment locality of the base station and MEC point. The deployment focused on making the distance between population and base station minimum. To solve transmission security issues in IoT devices, Gyamfi et al. [77] discussed using ECC-based scheme, which is considered lightweight solution. This proposed solution reduced the complexity and running time of traditional encryption algorithms, following philosophy of former ECC efficiency [108]. The paper [109] presented using identity-based anonymous authentication scheme for MEC anonymity and non-traceability. It also allows users whom already registered to access multiple MEC servers. Zhou et al. [110] investigated the security of MEC system in Unmanned Aerial Vehicle (UAV), which has multiple ground users. The paper also attempted to maximize the user's secrecy, minimize latency, efficient energy consumption, and minimize offloading requirements.

He et al. [111] analyzed the security issues presented in the IoT applications, which supports the MEC concepts. The IoT-MEC applications are perception systems and networked

Table 2 Comparative analysis of existing solutions for AI and FL based MEC

Paper	Year	Aim of the work	Proposed approach
Lillicrap et al. [56]	2015	Efficient management of high dimensional action space	Deep Deterministic Policy Gradient (DDPG) based agent can learn competitive policies
Sun et al. r-[57]	2016	Reduce traffic load in IoT devices	Edge-IoT architecture efficiently handles raw data streams and reduces traffic load
He et al. [58]	2017	Study cache-enabled opportunistic under time-varying channel coefficient and optimize the cache-enabled wireless network	Applying Data Deep Reinforcement Learning (DDRL) to obtain the optimal Interference Alignment (IA)
Guo et al. [59]	2017	To improve the local caching system performances	Applying Q-learning to dynamically replace the files in the cache of the Base Stations
He et al. [60]	2017	To investigate address location and usage pattern privacy in wireless offloading of MEC	Scheduling algorithm which also performs task offloading securely based on CMDP framework
Huang et al. [61]	2018	The optimize Quality of Experience(QoE) for multimedia traffic control based on SDN	Data Deep Reinforcement Learning (DDRL) based network traffic control architecture
Zhang et al. [62]	2018	To create an efficient private FL scheme and Maximize the total cache utility	To group linear model to accelerate reinforcement learning and create a mechanism that perturbs the client-side's Laplacian random noises
Bhagoji et al. [63]	2018	Demonstrate the vulnerability of FL Models with respect to poisoning attacks	Sophisticated detection strategies at the server
Bhowmick et al. [64]	2018	Investigate Model fitting under local privacy and difficulties associated with local differential privacy	Mini-max optimal privatization mechanisms
Fung et al. [65]	2018	Investigate Vulnerability of FL to Sybil attacks	Fools-Gold adapts to the the learning rate of clients, which is based on contribution similarity
Ma et al. [66]	2018	To manage increasingly fine-grained and complicated sensing data	Basic Privacy-Preserving Reputation Management (B-PPRM) and Advanced Privacy-Preserving Reputation Management (A-PPRM) schemes simultaneously
Zhang et al. [67]	2018	Provide the security to MEC architechture	Secure architecture called Vehicular Ad-hoc NETwork (VANET) that combines blockchain and MEC
Syamkumar et al. [68]	2018	Identification of Geographic Location for cell tower and data centers	Incremental deployment model based on results from Voronoi cell-based analysis
Li et al. [69]	2018	Design a MEC platform that can be easily deployed in 4G LTE and reference for 5G network	Middlebox approach is adopted to develop the MEC platform
Nasr et al. [70]	2019	Investigate privacy vulnerabilities of the stochastic gradient descent algorithms	Prove vulnerability to white-box membership inference attacks
Xu et al. [71]	2019	Perform load balancing and maintaining data integrity along with reducing offloading time and energy consumption	Designed BeCome model for dealing with data integrity and offloading
Gai et al. [72]	2019	Detecting Privacy-preserving problems in Smart Grid Networks (SGN)	Permissioned blockchain system to join all entities in Smart Grid Network (SGN)
Xu et al. [73]	2019	Investigating offloading problem considering the implementation utility	NSGA-III is used to create an offloading strategy followed by two primary metrics to be optimized simultaneously
Pang et al. [74]	2020	Investigate issues regarding user task offloading decision and resource allocation	Collaborative scheduling strategy
Xu et al. [75]	2019	To investigate the time consumption problem	Preserve user privacy with the time-efficient offloading method
Hui et al. [76]	2019	To investigate the resource allocation problem of deploying MEC-IDS in MEC environment	Proposes an allocation mechanism by using mathematical modelling

Table 2 continued

Paper	Year	Aim of the work	Proposed approach
Gyamfi et al. [77]	2019	Investigate transmission security in IoT devices	Novel lightweight Elliptic Curve Cryptographic (ECC) based solution
Huang et al. [78]	2019	To quantify security overhead experienced by a task in heterogeneous edge servers	Using a Scheduling Strategy named SEECO, which ensures security and efficient energy consumption
Ranaweera et al. [79]	2019	To identify several threat vectors of MEC	Intrusion Detection System (IDS), data encryption, Trusted Platform Manager (TPM), and Hypervisor self-examination methods are used for MEC security
Huang et al. [78]	2019	Investigate security and cost-aware offloading problem	Security and cost-aware computing offloading (SCACO) strategy based on a deep Q-network (DQN)
Truex et al. [80]	2019	Investigate extraction attacks and collusion threats	Combines Differential privacy (DP) and secure multiparty computation (SMC) to improve the accuracy and security of the system
Lu et al. [81]	2020	To stop data leakage in Vehicular Cyber-Physical Systems (VCPS) during the learning process	A two-phase Random Sub Gossip updating Scheme
Bagdasaryan et al. [82]	2020	Vulnerability identification in FL Model which can lead to Backdoor installation	Novel model replacement technology has been created to perform backdoor installation which also demonstrate it's efficiency
Yu and Li [83]	2021	To achieve better elasticity and resource utilization	Neural-structure-aware resource management technique
Feng et al. [48]	2021	A tradeoff between accuracy and training efficiency	A joint optimization algorithm
Liu et al. [84]	2021	Secure the machine learning model during data collection	Asynchronous convergence model considered staleness coefficient and a blockchain network for aggregation of global model
Ali et al. [85]	2021	Distribution of capabilities and offloading tasks of MEC	Deep reinforcement learning based multi-user context-aware offloading scheme

drones. They suggested a novel method for taking control of an autonomous non-cooperative Drone. In the MEC environment, due to several heterogeneous edge servers, security overhead may rise. To quantify security overhead with workflow scheduling problems, a secure and energy-efficient scheduling strategy has been projected in [78]. Similarly, Truex et al. [80] proposed a Privacy-Preserving framework in the FL environment. The aimed approach used hybrid modeling which combined differential privacy and secure multiparty computation to protect the system against inference attacks and collusion threats. Ranaweera et al. [79] have presented the current status and several threat vectors of the MEC paradigm from a security point of view. They discussed the use of Trusted Platform Manager and Virtual Machine Inspection for countering the virtualization-based attack. Relatively, a bio-surveillance framework for detecting multiple health security threats with the support of MEC has been proposed in [112] as framework, presented innovative techniques for collecting and representing monitoring information. A novel security architecture for Integrated Clinical Environments has been introduced as Integrated Clinical Environments that manages security, privacy, QoS, resources allocation, low latency, and high availability solutions in considerable manner.

Nilsson et al. [113] evaluated the FL algorithms such as Federated Averaging, Cooperative, Federated Stochastic Variance Reduced Gradient on the MNIST dataset using Bayesian correlated t-tests. Ahmad et al. [114] pointed out the main security issues in 5G, which, if not properly addressed, can become threatening. Moreover, the paper also presented potential security mechanisms and solutions for the discussed threats. Lu et al. [81] propose a sub gossip updating scheme based on FL to alleviate data leaks in the VCPS system. Cheng et al. [115] proposed a novel lossless privacy-preserving algorithm. They also used Secure-Boost to train a high-quality tree boosting model. In this approach, training data remains secret over multiple parties, similar in principle to increasing participants using counting-based secret sharing via involving matrices and practical steganography [116]. Bissmeyer et al. [117] conferred optimal secure mechanism in 5G-MEC architecture to deal with decentralization, security, location awareness, and minimal rejection. Hou et al. [118] proposed Access Control Mechanism with better protection for data in MEC architecture. The deliberated Fine-Grained Access Control mechanism considered user grouping to deal with the problems in access control policies. Huang et al. [78] examined the security and effective offloading issue framework as based on Markov decision process. They suggested security and cost-effective offloading strategy based on deep Q-network process, i.e., to find the best offloading policy. The paper's primary goal is to reduce total costs while adhering to the risk rate constraint in MEC. Belli et al. [119] proposed the use of Mobile crowd-sensing in scenarios where it's required to have massive sensing. They also analyzed real-world datasets. Mohri et al. [120] planned novel framework for FL based on principle learning objectives in which they presented a detailed analysis and learning algorithm.

Wang et al. [121] studied the allocation problem in MEC servers via two approaches, namely flat and hierarchical deployment. The paper presented hierarchical deployment as one of the approaches that can reduce response time than flat deployment. Elgendy et al. [122] proposed an offloading model which performed resource allocation and computation offloading efficiently in a multiuser MEC system. The model also used Advanced Encryption Standard to prevent leakage of sensitive information.

In work [123], edge computing-assisted FL framework is anticipated in which both the training efficiency and accuracy are computed. In many realistic applications, AI algorithms are computationally expensive task requiring large-scale training samples. The huge amount of node deployment needed another concept that improves the efficiency of the system. Relatively, Machine Edge Learning (MEL) has been proposed in [124]. In this MEL concept, computationally expensive algorithms are carried out in the nodes or edges. Each node performs its own training iteration to train its local training model. After that, the local node sends the results to the higher-level nodes, aggregating the local features and sending updates to the lower-level nodes. A more intelligent AI-based edge system has been given in [125]. In-Edge AI framework Deep Reinforcement Learning techniques and FL concept is introduced with the mobile edge devices. This approach reduced the computing and communication load from the edge networks. To address the corresponding problems, research [126] suggested framework that combines FL and MEC. The work used open-source dataset CIFAR10 [127–129] for experimentation purposes compared with centralized learning.

Paper [130] presented FL as a Service (FLaaS) arrangement, which enabled 3rd-party applications to create ML models which are cooperative, decentralized, and preserve the privacy of data. An FL chain model deployed on a blockchain network composed of edge devices has been proposed in [21] to improve the security in FL. The paper projected the use of separate channel for learning global models in the blockchain network in a smart way.

A black-box- and white-box-based resource optimization approaches in federated learning have been discussed in [83]. After that, a neural-structure-aware resource management technique was proposed for better elasticity and resource utilization. In this approach, each mobile client is assigned different working subnet based on the status of their local resources. A joint optimization algorithm has been briefed based on designed optimization problem [48] in federated learning-based MEC systems. This algorithm provides a tradeoff between the accuracy and training efficiency of the model. In an edge computing environment, data collection for machine learning algorithms raised many security and privacy issues. To solve this problem, Liu et al. [84] proposed an asynchronous convergence model in federated learning. This technique considered the staleness coefficient on blockchain network for aggregation of the global model. A multi-user context aware offloading scheme has been developed in [85]. They also use deep reinforcement learning (an FL based model) for the capability distribution of MEC devices. A privacy-preserving framework has been planned in [131] to protect the system from data leaking and privacy issues. The framework considered both federated learning and edge computing environment with deep learning model, so that data can reside locally on edge devices and end users. Inference attacks are considered for privacy analysis of edge-FL-based environments. In the federated learning model, malicious nodes can upload fake/unreal learning parameters, giving high error rate. A Federated learning parameter aggregating algorithm has been briefed in [132] to resolve such problems. The mutual information will be used to calculate the similarity of the gradient trend between local training model and overall model. Asynchronous federated learning approach has been developed in [133] to manage the synchronization optimization. This model allows the edging node to select some part of the model, which will reduce the amount of calculation and communication. Resulting from this, the model efficiency has increased in heterogeneous edge environments.

3 Background of MEC

In this section, the background of the MEC for IoT is discussed. The technical development, characteristics of MEC, MEC technologies, MEC actors and their roles, MEC access technologies, the objective of MEC, network architecture, and advantages are some points on which the section is developed.

3.1 Deployment and Technical Development

The first concept for the development of Mobile Edge Technology is introduced in [7]. In this concept, Cloudlets is to be used as computing "hotspots" similar to WiFi hotspots. Another WiCloud architecture is discussed in [134] on which the MEC has been developed. These are the foundation blocks for improving technology. European Telecommuni-

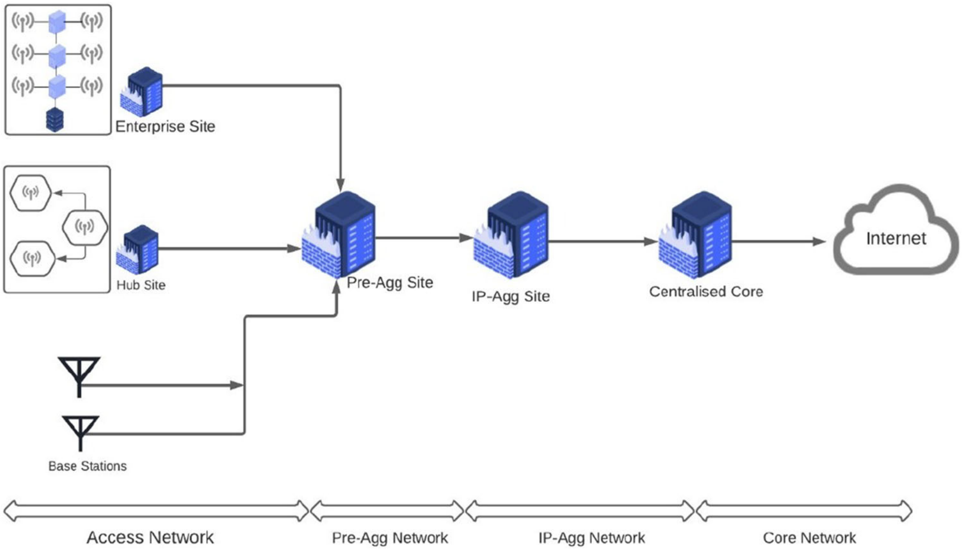

Fig. 2 Mobile edge computing development architecture

cations Standards Institute (ETSI) is the first organization that standardizes the MEC network architecture by integrating cloud computing and IT-enabled services [8].

Nowadays, it is combined with other technology such as 5th generation 5G networks. MEC infrastructure elements can be deployed in multiple places in the network depending on the use. The MEC servers are located at multiple locations. For instance, in an LTE cellular network, Marco Base Station (ENodeB) is used to deploy the application server. In a 3G cellular network, Radio Network Controller (RNC) [135, 136] can be used to deploy the MEC server. The pre-existing 4G network system is divided into four parts: access network, pre-aggregation network, IP-aggregation network, and core network [137]. The access network is further divided into enterprise site. The pre-aggregation network deals with processes that have low computational requirements. The IP-aggregation network differentiates traffic based on the service which is requested. Finally, the core network deals with the high computational service and analytic. A network system for the development of MEC is shown in Fig. 2. The new 5G technologies are composed of virtualization [138, 139], the programmability of networks and services that is under development with 3rd Generation Partnership Project (3GPP) [140, 141]. 5G is a revolutionary technology that may solve many problems faced by 4G technology problems, such as higher bandwidth and lower service latency [142].

3.2 Characteristic of MEC

MEC technology composed several computing platforms like fog computing, cloud computing, mobile computing, IoT, wireless technologies, and many more. Due to this, it contains numerous characteristics. ETSI white paper states the following as the main characteristics of MEC.

- *On Premises:* MEC platforms are not dependent on the underlying network architecture. During accessing local resources, they have separated from the other networks. This property of MEC makes it less vulnerable as the MEC network is not centralized and not dependent on any other networks.
- *Proximity:* Most of the time, MEC servers are being deployed closed to the accessing points so that the computation and transmission time may reduce. This feature makes the system more capable of handling big data applications with large data size and low computation time.
- *Low Latency:* MEC servers are deployed at closed proximity of the user devices, and data movement is separated from the core network. Due to this, the service latency and communication delay may reduce. This leads to higher bandwidth and better QoE.
- *Location Awareness:* The MEC technologies are mainly based on the machine-to-machine concept. In such a scenario, devices use low-level signals for information sharing. MEC uses low-level signals to discover the device location identification.
- *Network Context Information:* The MEC provides real-time network information for the implementation of real-time business applications. Based on RAN [143, 144] information, people can estimate future behavior and congestion of the network. This will also help to make smart decisions for better QoS delivery.
- *Dense Geographical Distribution:* The MEC components are distributing among multiple geographical locations. Each movable user can access the services at the edge of the network in which the edge network is fixed, and the user is located at multiple locations [145].

3.3 MEC technologies

MEC technologies and modeling components can be briefed as follows:

3.3.1 Local Cloud

To ensure data privacy and lower latency, the local cloud is used in the local network connection with the remote cloud server. In most cases, the software is installed on the local cloud and integrated with the remote cloud. This will make the system fast and reduce the communication delay. But, the local cloud has limited capabilities [146].

3.3.2 Cloudlets

It is a small-scale database that is generally located at the edge of wireless hop. This is located near to end mobile users or devices. The Cloudlet [147, 148] is connected with the remotely located cloud server to provide the services efficiently. The primary focus of cloudlet is to reduce the distance between end mobile users and installed service locations. This will reduce the service latency and energy consumption for latency-sensitive applications.

3.3.3 Fog Computing

Fog computing [149–151] or Edge Computing was created by CISCO that ubiquitous connected devices at the edge network. This computing technology carried out network service and computing resources in LAN near IoT gateway or fog node. This provides lower latency when compared to cloud computing.

3.3.4 Virtualization

Virtualization is a converting technology in which physical IT resources are converted into vitalized resources. Virtual servers can create a virtual disk image that contains the backup file of the virtual server as virtual logical resources in the same physical hardware. All the MEC services and resources such as memory, storage, network infrastructure, power, operating systems are accessed through the virtualization concept. The MEC provider creates multiple virtual resources at the edge layer.

3.3.5 High Volume Servers

The MEC contains several high-volume servers deployed on the edge network. The responsibility of the server is to perform network traffic forwarding and filtering task. It is also responsible for executing the offloading task.

3.3.6 Network Technologies

The MEC is composed of several small–small computing devices and network technologies. Multiple mobile nodes, sensors, wireless stations, edge servers, computing devices are the key components of MEC.

3.3.7 Mobile Devices

Mobile devices are the main component of any MEC infrastructure. Its main feature is portability. It can perform low computationally intensive and hardware related tasks which relieves some load from the edge server. Portable devices also perform P2P computing within the edge network through D2D communication.

3.3.8 Software Development Kit

With the help of standard Application Programming Interface (API) [152] and software development kit, anyone can develop new edge applications which are easily adaptable and integrated with the current MEC applications.

3.4 MEC Actors and Their Roles

The following are the main actors of MEC, along with their roles.

- *Application Developer:* Applications developers design MEC applications that are used for accessing the MEC services. They may create custom MEC software for a specific MEC customer or commercial software sold to the general public.
- *Content Provider:* A content provider takes material and prepares it for distribution in the network. Its main responsibility is to collect all the local and real-time information about the network and distribute it among the participating nodes.
- *Mobile Subscriber:* Mobile subscriber is the actual users who are subscribed to the MEC services. They have accessed all the MEC services through portable mobile phones.
- *OTT Players:* Over-the-top (OTT) players share or receive the television or video materials as a standalone product via the Internet. Instead of using traditional methods like cables, OTT providers deliver video content over the internet.
- *MEC service provider:* MEC service providers can utilize their network resources and introducing new innovative applications and services in front of MEC users that require low latency.
- *Software Vendors:* An independent software vendor, also known as a software publisher, is an organization respon-

sible for making specialized MEC software and sell it to other customers. They have not concerning about computer hardware components required in MEC.
- *Network Equipment Providers:* Network equipment providers (NEPs), sometimes called telecommunications equipment manufacturers, are companies that sell products and services. The product and pieces of equipment are required for network communication.

3.5 MEC Access Technologies

The MEC network's deployment at the mobile network's end allows current mobile infrastructure services to be optimized. In the LTE downlink, Mobile Edge Scheduler reduces the average latency of general traffic flows [153, 154]. MEC aims to deploy multiple servers in proximity to deal with latency issues. However, deploying physical servers will be very expensive. To deal with this situation, several virtualized servers can be deployed on multiple mobile networks. Some deployment locations considered by the MEC ISG are LTE/5G base stations (eNodeB) [155, 156], 3G Radio Network Controllers (RNC), or multi-Radio Access Technology (3G/LTE/WLAN) [157, 158] cell aggregation sites. The MEC ISG has suggested that this virtualization infrastructure should not only limit itself to MEC services. Related services such as NFV [57, 159] and SDN [160, 161] shall also be hosted in the virtualization infrastructure. Moreover, Bluetooth is another access technology to communicate between devices.

3.6 Objective of MEC

The following are the objectives of MEC:

- *Minimizing Latency:* With the servers being placed close to the user end devices, the communication time is reduced drastically.
- *Minimize Energy:* By offloading high computational work to the nearest edge server relieves the user devices from intensive work. This also helps in reducing the energy consumption in the user device.
- *Minimize cost:* The cost of hardware and software components can be reduced by deploying MEC on the virtualized servers.
- *Maximize Throughput:* In the MEC architecture, everything is processed near the edge gateway. A short communication distance between the server and end-user can maximize the throughput of the MEC network.
- *Minimize Radio Utilization:* Using MEC, a chunk of network tasks will be pre-processed at the node server and solve some of the network requests at the edge server. This would reduce the traffic reaching the cloud server. This can produce good results in terms of fast communication.
- *Optimize Computational Resources:* By offloading the resource-intensive work to the edge server and performing low resource-oriented works in the user devices, the MEC network optimizes the computational resources.

3.7 Network Architecture

The first MEC reference architecture was developed by ETSI MEC ISG in 2016 [162]. The network architecture of MEC contains cellular network communication system infrastructure known as RAN provides communication between wireless controlled devices (mobile phones, sensors, cellular radio system). Some of the network architecture of MEC under different scenarios are found in [35] and [134]. The base concept of each MEC network architecture uses IT and cloud computing capabilities at the edge of the mobile network. This feature supports low latency, high bandwidth and enhances the performance of the system. It mostly resides between mobile users and the cloud. The network architecture of MEC is shown in Fig. 3. It contains a three-layer network architecture–User/system, Mobile edge, and Enterprise/core. It also contains several Base Stations (BS) that provides high radio coverage. The various wireless interfaces enable distributed BS to collect data from multiple edge devices, whether moving or stationary. The first user/system layer consists of edge computing devices and edge device management services. The management service will help run edge applications inside the mobile device under an operator network. This layer provides an interface to mobile edge hosts for accessing the edge computing service via mobile edge application.

The second layer is the mobile edge layer which receives all the edge traffics generated by edge devices. It contains an edge platform and virtualization infrastructure that handles the management of edge specific computing tasks. All the computing tasks are performed with the help of geo-distributed physical or virtual servers with built-in IT and cloud computing services. The servers and computing devices are deployed near to the mobile users and use cellular network capabilities. It also performs less resource extensive analytics and store frequently accessed data in the cache. At the enterprise/core layer, the user can store their data in cloud and database servers. The user can perform high computation resource-intensive analytics and optimizing operations. When the edge nodes do not have adequate computational resources to handle their local data, they can offload their computing task to the cloud by adding more network resources and higher service latency. The addition of this layer makes the system resource-rich and increase the battery lifetime of user devices.

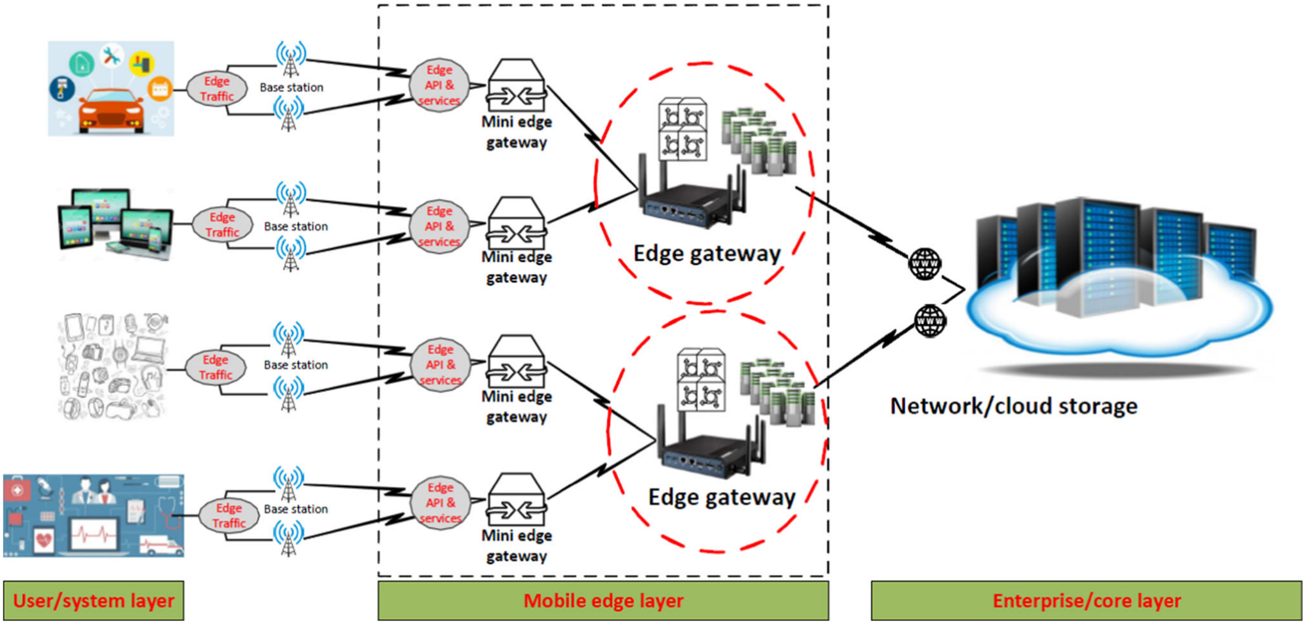

Fig. 3 MEC network architecture

3.8 Advantages of MEC

There are several advantages of MEC. Their advantages are just not limited to the user end but also benefited to Mobile Network Operators (MNOs) [163, 164], Application Service Providers (ASPs) [165–167], Over-the-top (OTT) Players, and many more entities. Some of the advantages are stated below [168].

- *Reduced communication delay:* Edge Computing aims at storing information at the close of the mobile edge server. This kind of data localization reduces computational complexity. It also reduces access delay with respect to latency. Also, the network bandwidth increases as lesser resources are needed to transfer data. Frequently requested data are stored at the node database, which reduces the communication overhead on the network.
- *Aggregation:* MEC Servers are capable of aggregating similar or related traffic. These results lead to less network traffic and a positive impact on bandwidth utilization, scalability, and power consumption. Aggregation also helps in monitoring similar types of data from various devices that are aggregated together.
- *Augmentation:* With the augmentation concept, more information is available at the base station. This data can be analyzed statistically and shared with the ASPs to provide better QoE. MEC-based augmentation comes with low network delay because ASPs can adapt service parameters in real-time.
- *Deploying application:* MNOs can enable Radio Access Network (RAN) in a distributed computing environment to deploy applications and services. The exposer of RAN elements and information makes the deployment of applications and services more accessible and flexible. Enabling these services could help generate more revenue. Services like excess storage, speed, computational resources can be charged.
- Infrastructure as a Service (IaaS) Platform: By enabling MEC enabled IaaS Platform [169–171] at the Network Edge, ASP services can be scaled along with higher bandwidth and lower latency. If ASPs could get real-time access to radio activities, it can lead to better application development.
- *Computation offloading:* By offloading resource-intensive processes to the nearest edge node provides better QoS. High resource-intensive works cannot be performed on the user side due to limited hardware capabilities. This type of Offloading is performed if the power consumed for computing is more than for wireless transmission.
- *Balancing workload allocation:* Geo-distributed local authorities can work together to process healthcare data synchronously. Adding a load balancer in the Edge Layer gives more control and a balanced workload of the global information at the edge nodes.
- *Security:* MEC can perform the new level of surveillance and monitoring using video analysis. This analysis can be performed on the edge nodes. Also, the data can be received by the decision-makers very fast. The edge nodes are generally close to the users. The proximity of encrypted and signed traffic makes it more secure.
- *Easier management:* The nodes are managed from the management hub. This makes it easier to implement secu-

rity patch updates, changing functionalities, and many more. Moreover, it also makes it to find out the node that is not functioning properly.

4 FL Approaches for MEC

This section discusses the basic conceptual architecture of federated learning and the FL approaches/technologies used in MEC environment.

4.1 Overview of FL

FL allows mobile devices to learn from a prediction model cooperatively without sharing their local data, differently than secret sharing of [172]. It is also known as collaborative learning. This machine learning approach trains an algorithm with local data samples stored on the decentralized edge devices or servers. There are several open-source FL frameworks are identified [173–176]. Traditional machine learning approaches required a centralized local training dataset stored on one machine or one server. The difference between FL and other distributed learning schemes is that the local data are not exchanged between edge devices in FL. In contrast, in other learning approaches, the data are distributed in the environment. FL is considered a local dataset that resides on a single user's end device and does not represent the overall population distribution.

The local datasets generated across federated learners may differ greatly in terms of size because they are independent of one learner device to another learner device. This can lead to an imbalanced distribution of the data across multiple nodes.

4.2 Formulation of AI and FL Models

In real-time MEC environment, there are multiple mobile edge nodes denoted as:

$M = \{m1;m2;m3; \ldots\ldots;mn\}$, where n is the number mobile edge nodes. Each of mobile edge nodes are participated in training a local model and sharing of that model with the help of their own database $D = \{d1;d2;d3;\ldots..;dn\}$. In the learning scenario, no edge devices can directly access the data from other devices. In every communication round, each edge devices train a local model and compute an updated *wn* with the local data dn. In this process, the edge device mn does not require to share their data to other devices. The total learning sample size is $\sum_{n-1}^{N} = n_k$; where n_k is the number of samples. The federated learning problem can be defined as minimizing the risk from the learning model. The mathematical formulation of federated learning is represented by using Eq. 1.

$$\underset{w\in R^d}{min}\, LF(w) = \sum_{n=1}^{N} \frac{n_k}{n} LF_k(w)\,;\; where\; LF_k(w) = \frac{1}{n_k}\sum_{x_i\in d_n} lf_i(w) \tag{1}$$

Note that w is the model learning parameter. The function *lfi(w)* is computed with the help of loss function. The value of loss function should be minimum for a good result. It is dependent on input–output data pair $\{pi, qi\}$. Where $pi \in R^d$ and $qi \in R$ or $qi \in \{-1, 1\}$. The mathematical representation of loss function varies from algorithm to algorithm [45, 177]. Mathematically, the loss function is represented by Eq. 2, as detailed in [53]. The loss function of few standard learning models is represented by Eq. 3 (Linear regression), Eq. 4 (Logistic regression), and Eq. 5 (Support vector machines).

$$W_n^* = \text{argmin} LF(w_n) \tag{2}$$

$$\text{Linear regression}: lf_i(w) = \frac{1}{2}(p_i^T w - q_i)^2, q_i \in R \tag{3}$$

$$\text{Logistic regression}: lf_i(w) = -log\left(1 + \exp\left(-q_i p_i^T w\right)\right), q_i \in \{-1, 1\} \tag{4}$$

$$\text{Support vector machines}: lf_i(w) = -\max\{0, 1 - q_i p_i^T w, q_i \in \{-1, 1\} \tag{5}$$

After *training* of local models, all the models are uploaded to the server. The server aggregates all the received local models $w1$,$w2$,….wn to make a global model Wg. The global model is updated or downloaded in each of the edge node and then replaced the local model. Now, the new global model is used for training purpose in next round until the global learning process is completed. A diagrammatical representation of federated learning is shown in Fig. 4.

4.3 FL Approaches/Technologies used in MEC Environment

This *subsection* summarizes the FL approaches/technologies used in MEC environment. The following aspects are considered to realize its underlying technology [178, 179].

Data partition in FL is helpful to build ML applications in which data are kept private throughout the training process.

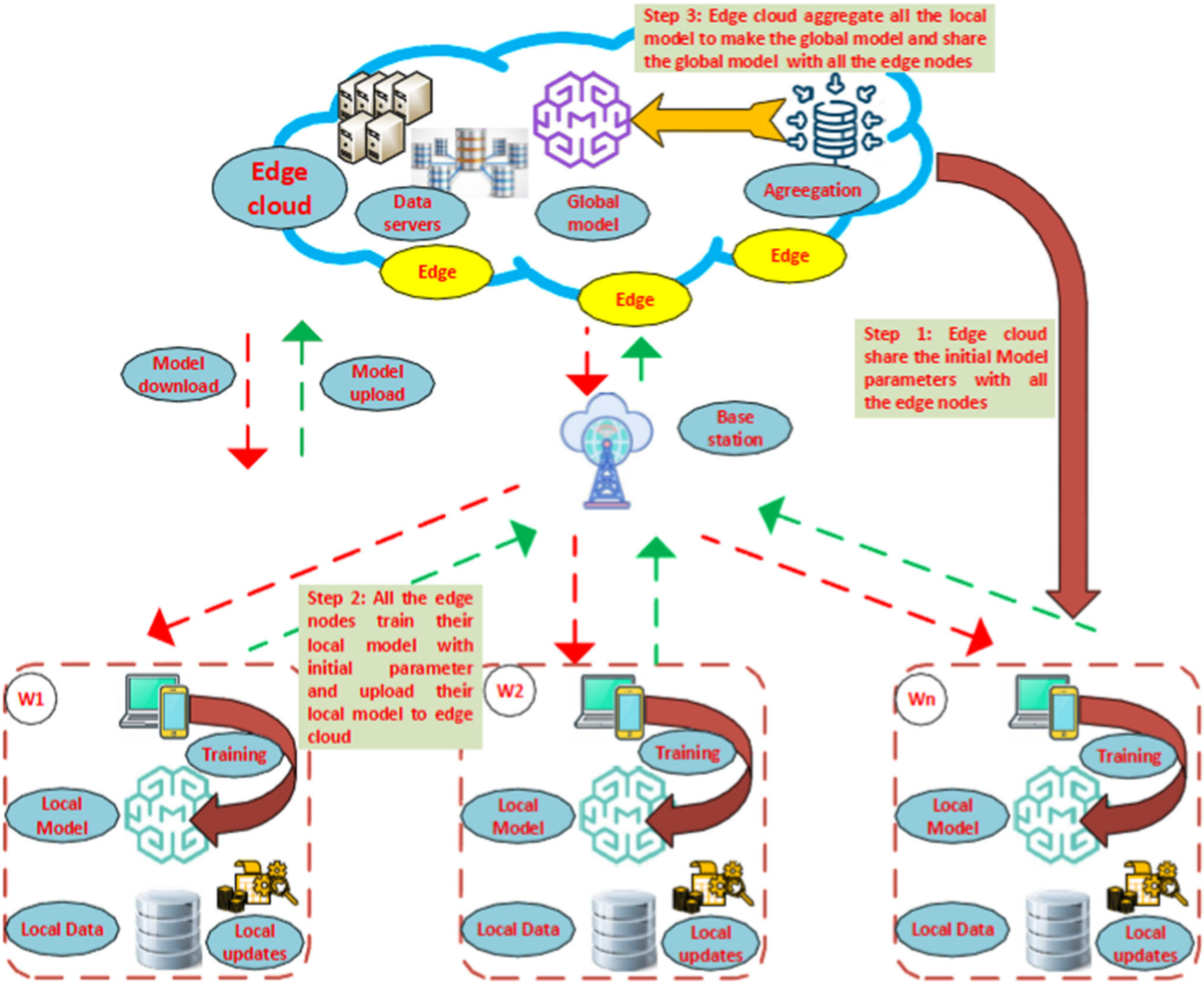
Step 3: Edge cloud aggregate all the local model to make the global model and share the global model with all the edge nodes
Edge cloud
Data servers
Global model
Agreegation
Edge
Edge
Edge
Step 1: Edge cloud share the initial Model parameters with all the edge nodes
Model download
Model upload
Base station
Step 2: All the edge nodes train their local model with initial parameter and upload their local model to edge cloud
W1
W2
Wn
Training
Local Model
Local Data
Local updates
Training
Local Model
Local Data
Local updates
Training
Local Model
Local Data
Local updates

Fig. 4 An architecture of federated learning in MEC environment

There are three different approaches in FL for the data partition. These are horizontal partition, vertical partition, and federated transfer learning.

Horizontal FL is also known as sample-based FL, is applicable when the dataset samples share the common user attributes or features, but the sample belongs to different users or datasets. This dataset is divided horizontally by consideration of user dimension and user characteristics. It does not consider the same users while data are split. The most famous example is Gboard which is Google's keyboard on Android.

Another example of the medical sector in which medical researchers uses machine learning models to determine the possible occurrence of cancel cells. In such applications, each participant will get a new model by sending a gradient to the server. The server aggregates all the local gradient to make the global model that will be helpful to train the individual device. During the process, the private node information may leak. The standard solution to protect the data leaking is secure aggregation, homomorphic encryption, and differential privacy [180].

Vertical FL is often used when the two datasets share the same sample ID (user ID), but the feature of the dataset overlap little. The data is divided vertically based on the user ID and take the part of the data in which users are common, but features of the dataset are different. In this technique, the number of features for the training may increase. There are many machine learning models which use vertical federated learning. Some of the models are logical regression, classification, safe linear regression, statistical analysis, and data mining. It is an excellent exciting AI technology that can provide better personalized MEC services without compromising user's privacy.

Federated transfer learning is a classical learning process in which both users and user features of two datasets rarely overlap and data is not segmented. In this technique, the process used a pre-trained model that is already trained on similar datasets to train a new model. The pre-trained model is an already trained model for solving of an entirely different problem. The assumption is a pre-trained model gives much accuracy compared to a trained fresh model built from scratch.

Privacy mechanism presented in 2017, as Google was the first organization that supports a federated learning approach for privacy-preserving in machine-learning models. The most important feature of Fl includes the raw data of each edge node is stored locally without exchanging or transferring to other nodes.

Model aggregation is a federated learning method that provides privacy solutions. This process trained the global model by the combination of multiple local model parameters received from multiple nodes. Shashi et al. [181] defined an incentive system that allowed numerous devices to participate in training the model.

This will achieve effective outcomes and improve communication efficiency. Yu et al. [182] showed a local adaptability model based on fine-tuning, multi-task learning, and knowledge extraction. In this model, individual participant privacy and the benefits of federated learning both can be achieved.

Homomorphic encryption is the traditional encryption techniques are the most widely used data security solution. Users without a key cannot extract plain text information from the encrypted data in these data encryption techniques. It means the security strength entirely depends on the key and without the key, the decryption is failed. The homomorphic encryption mechanism resolves the key's computation issue by focusing on data processing security rather than the key. It enables arithmetic operations on encrypted data. This is also known as secure multi-party computation. The feature also allows users to encrypt their information in such a manner without opening the original data or without decryption users can calculate and process the encrypted data. Partially Homomorphic Encryption (PHE), Somewhat Homomorphic Encryption (SHE), and Fully Homomorphic Encryption (FHE) are the three categories of homomorphic encryption. PHE schemes support only one single arithmetic operation on ciphertexts. It is categorized into two different techniques: additive and multiplicative. Paillier cryptosystem is an example of additive PHE. RSA and ElGamal is an example of multiplicative PHE, which are proven not preferred compared to ECC [183]. Hardy et al. [184] presented a federated logical regression model that employs an additive homomorphism method to protect the system. Liu et al. [185] developed a federated learning framework for transfer learning in which the privacy mechanism additionally employs using additive homomorphic encryption to encrypt model parameters.

Differential Privacy is a possible privacy concept suggested by Dwork in 2006 to address the issue of privacy exposure in statistics datasets. According to this definition, the database's calculation results are unaffected by changes to a single or specific record. Even the dataset has minimal influence on the calculation results. As a result, the risk of privacy disclosure by looking at or modifying a record into the dataset is minimum. The attacker is also unable to acquire precise individual information by looking at the calculated results. The training process of machine learning and deep learning includes noise in the output to apply differential privacy in gradient iteration.

Data availability is the process of ensuring that data is available to end-users and applications when they need it is known as data availability. It refers to the accessibility and continuity of information. The FL approach is divided into two categories based on the availability of the data and the number of edge nodes.

Cross-silo FL affects Edge nodes in this scenario which are often small-scale with a cluster of 2 to 100 devices. Training data are classified into horizontal learning and vertical learning. Cross-silo FL is more versatile and easier compared to cross-device FL. It is used within organizations or groups of organizations to train the ML model with their sensitive data. The encryption scheme is used to secure the information from the client as well as from the attackers.

Cross-device FL is the technique that contains a large number of edge nodes that belongs to a similar domain with similar interests from the global model. Due to the high number of users, it is tough to keep track of all nodes and preserve the transaction history of the records. Clients frequently connect across untrusted networks where node selection/participation in training rounds is totally random [186].

Aggregation run on the algorithms that help FL reach the goal global ML model by binding updates from multiple nodes. This logic must be configured the node heterogeneity, variable weights of each local model, and communication problems. FedAvg, SMC-Avg, FedMA, FedProx, Scaffold, Tensor Factorization, and FedAttOpt are some of the aggregation algorithms. The general federated learning aggregation scheme always uses at least two layers of aggregation: Local on-device aggregation and cross-device (or federated) aggregation.

Learning models are federated learning is facilitated with popular machine learning models whose main aim is to ensure the model's privacy, accuracy and efficiency. Linear model, Decision tree model, and Neural network models are three popular ML model supported by FL.

Du et al. [187] proposed a security solution that addresses the security challenges of entity parsing. The security solution is developed in the federated environment to train a linear model. They achieve the same accuracy as the non-FL approach. Nikolaenko et al. [188] created a ridge regression system using homomorphic encryption. The linear model is simple to apply in comparison to other models. Thus, it is an excellent model for adopting federated learning.

Decision trees such as Gradient Boosting Decision Trees (GBDT) and random forests can be trained via federated learning. The GBDT method has received a lot of attention in recent years because it performs very well in the case of classification and regression. Zhao et al. [189] use the GBDT privacy protection system in regression and binary classification tasks. The system securely aggregates regression trees built by multiple data nodes to prevent the exposure of user data privacy. Cheng et al. [190] introduced SecureBoost. This framework allows users to create a federated learning system by training the gradient lifting decision tree model for horizontal and vertical division data.

Neural network models provide smart AI solutions with data privacy and security. It trains neural networks to improve the efficiency of the application, maintain privacy within the system, and complete complicated tasks. Deep neural network-based Drones application can help to build trajectory planning, target recognition, and target placement. The Unmanned Aerial Vehicle (UAV) group usually trains the model through deep learning to provide efficient services. The centralized training method cannot play the UAV's real-time performance due to the absence of a constant connection between the UAV group and the ground base station. Zeng et al. [191] discussed a distributed federated learning algorithm applied to the UAV group for optimization of federated learning convergence speed and perform joint power allocation and scheduling. Liu et al. [192] propose a clustering FedGRU method that achieves the best global model and captures the Spatio-temporal correlation of traffic flow data. The model performs more accurately by combining the Gated Recurrent Unit neural network for traffic flow prediction with federated learning. Experiments on real data sets demonstrate that it outperforms non-federated learning approaches significantly.

Network topology is the arrangement of the edge nodes that link of communication network as a topology. Network topology is a term that may be used to describe or define the layout of many types of telecommunication networks such as command and control radio networks, industrial field buses, and computer networks.

Centralized and clustered FL is the base concept of FL built to serve decentralized data as strategy. But, still, it relies on a centralized server to manage the duty of gathering trained models from different FL edge nodes, building a global model, and sharing it with all edge nodes. This is mostly used to construct a third-party system to increase edge nodes confidence. The traditional centralized server hosts data and trains a given model on shared data. But, the centralized server in the FL environment works on a shared model via synchronous or asynchronous edge nodes updates. Gboard, an Android keyboard created using Tensorflow and federated from Google, is an example of a centralized FL method [83].

Fully decentralized FL are related to edge nodes in decentralized FL to work together as training model in a peer-to-peer way without the use of a server. Any edge node can start the training process by defining the model, loss function, and algorithm. After that, interested edge nodes can register and take part in the training. In decentralized FL, the model is split into many partitions replicated on different edge nodes. But, in centralized FL, only the server can store, modify, and broadcast the model to the participating edge nodes. Pappas et al. [193] implements a functional prototype for Interplanetary File System (IPLS) to measure its performance. For the simulation of the connectivity between the edge nodes, they use mininet. Each mininet node is an edge node that uses IPLS to participate in the training of a model.

Federated learning parameters are adopted once the learning network has been established, i.e., any edge node can choose different learning parameters to optimize the model. The number of federated learning round (K), the total number of participating edge nodes during the learning process (E), set of privileges used at each iteration for each edge node (P), and local batch size used at each learning iteration (B) are some essential learning parameters which can be changed over the time and changing of network scenario. Number of iterations for local training before pooling (L) and local learning rate (k) are model-dependent parameters.

Federated learning heterogeneity is a scenario for efficiency rating of the entire training process as affected due to the presence of heterogeneous communication devices.

In a traditional data-centric network, two popular communication techniques are available: synchronous communication and asynchronous communication. The consideration of multiple heterogeneous communication devices may easily disturb the synchronous communication method. The asynchronous communication strategy may better address the situation of multiple heterogeneous communication devices in the federated learning multi-device environment.

Fault-tolerant method can protect the system from collapsing in an unstable network environment, especially in a distributed environment. When many devices operate together, a device failure will have an impact on other devices. Federated learning is a promising solution that helps in such cases with maintaining of device security. Some of the research works [194, 195] not considered the device failure cases during the implementation. Thus, the system efficiency is not affected by the failure of any device.

Model heterogeneity is base foundation of any learning model as the sample data. The gathering of dispersed data from multi-party devices to train the federated model may affect the overall efficiency of the model. Thus, processing the heterogeneous data collected from different devices is important for maintaining global model efficiency. Multiple modeling solutions are available to handle the problem present in heterogeneous data. Some of the solutions are: If the device is single, their own model is the final model, develop a common global model that will apply to all the nodes, and train only specific models required for the tasks.

Communication efficiency is a federated learning approach assuming the complete learning process is distributed over multiple edge nodes. The entire communication load is figured as the summation of the total number of bits transferred from edge node to server or server to edge nodes by each client (C). The overall communication efficiency is computed with the help of Eq. 6. In this Eq. 6, U is the entire updates done by edge noes, $|S|$ is the model size, $E(\Delta S^{\text{upload/download}})$ is the entropy of the weight updates, and Υ is the difference between true and minimal update size. The complete update size is defined as: $|S| * E(\Delta S^{\text{upload/download}}) + \Upsilon$

$$\alpha^{\text{upload/download}} \in \partial\left(U * |S| * \left(E\left(\Delta S^{\text{upload/download}}\right) + \Upsilon\right)\right) \tag{6}$$

5 Security & Privacy Challenges with Countermeasures in AI-Based MEC

Security and privacy challenges include protection of interconnected systems and network devices from data theft, damage to hardware or software, loss of important data, misdirection of the service provided and many more. Any security mechanism aims to stop such kinds of issues/threats and maintain Confidentiality, Integrity, and Availability (CIA) [196–198] into the system. The discussion of security and privacy challenges with their countermeasures is summarized as discussed next.

The AI-based MEC is an emerging technology that has been rapidly growing in the last few years. Thus, it is required to study how much it is vulnerable to security threats from bad actors. AI is an intelligent programming technique to determine the pattern, predicting the values and outliers in the given datasets [199]. For this, the quality of the data should be maintained. The quality of datasets improves the accuracy of the AI model. However, in the case of AI-based MEC produce a high volume of data. The maintains and extraction of quality data is a challenging issue in the AI technique. Bad actors can feed polluted training datasets, which reduces AI accuracy, which is known as poisoning threats. The bad actors can provide some new inputs to change the output of the model. This can suffer from evasion threats. They can also customize the AI software component with public API, which is not much secure. The security of AI models and software components itself is a challenging issue. There need some solutions before deploying it to provide IoT security. Machine Learning techniques are also exposed to security vulnerabilities. In [200] inspects security concerns of outsourcing training of machine learning models and acquisition of these models from online model zoos. They identified several points of entry that can be used to introduce *backdoor threats*. They also identified several cases where maintaining the integrity of shared pretrained models is very difficult.

The main issue with AI-based MEC is that it is difficult to detect flaws in the system [201]. AI systems have a dynamic, networked, and adaptive nature. From the user's perspective, it is not easy to understand their internal process and behavior. The user also does not identify the resultant outcome as accurate or may change due to unwanted activities/threats. For instance, backdoor threats in the neural network may change the system's behavior if some malicious trigger has been activated, and identifying these threats/triggers is challenging.

MEC with AI also faces many security and privacy challenges [202]. The edge servers are vulnerable to *Denial of Service* attacks in which the attacker sends many false network packets to the edge server. This can cause unnecessary heavy traffic on the edge server. This attack may reduce the network performance or create a service availability issue. When sensitive data is offloaded to edge nodes, direct or physical control over the data may be lost. This can lead the data breaches and privacy issues of the data. Data storage can be audited using appropriate auditing procedures to ensure that data is stored correctly. Before storing the data to the MEC server, it must be encrypted. Secondly, 5G network providers should undergo external audits and security certification. Sharing of internal policies creates a loophole that may hamper the security of the system. A user's physical data's precise location is less transparent, confusing specific jurisdictions and commitments to local privacy requirements.

As data will be stored in a shared space, each user's data should be separated using encryption methods and a data-sharing mechanism. This also needs a proper multiuser fine-grained access control mechanism in which each user has different access privileges. Secure data encryption techniques may overcome the sharing issue because only authorize party knows the decryption key.

The MEC technology is leveraged several virtualized deployment models such as NFV, ICN and SDN. This virtualization infrastructure provides shared network resources among multiple users. The leaking of one resource information can affect the whole network infrastructure. The loss of data could completely deplete the resources used to perform computationally, storage, and network tasks and deny request services. Bad actors can misuse virtual resources. The VM sprawl threats happen when the resources administrator cannot control or manage the virtual network resources. It is also known as virtualization sprawl. In this case, bad actors can use some exploit running on the old OS, which has not been patched due to VM sprawl [203, 204]. Another security threat is design flaws. This is the error produced while configuring the system or due to insufficient security training. Hypervisor hardening [205–207], network abstractions [161, 208], and isolation policies are some security solutions that may overcome the security flaws and protect the virtual and physical server from different types of threats. An important security challenge arises while data are moved from the edge to the cloud servers or vice-versa. During the transmission, data may be intercepted or changed.

MEC also needs to offer proper recovery mechanisms in case of data loss. One solution is to create multiple backup files of the same data. In MEC, users' data are kept in a shared location. This makes investigating or searching data is a time-consuming process. The authenticity and privacy-preserving of the hardware and software components should be ensured. The encryption mechanism such as homomorphic encryption [209–211] can maintain the user privacy computation that has been performed without decryption of the original data. Secure authentication in a local MEC ad-hoc wireless network can help with Authentication and Identification problems [212]. For authentication purposes, the connected device uses authenticated key protocols and a Stand-alone authentication mechanism. It also has to be checked that wrong information does not result in wrong actions, resulting in loss of money, information, and user privacy. A secure transmission medium must be needed for exchanging information through a wireless medium. Intrusion detection systems (IDSs) [213] and Intrusion Prevention System (IPS) [214, 215] are some mechanism that will monitor the network packets and analyze the system traffics logs. This system will restrict any unauthorized access and generates alerts for malicious packets. This system feature protects the MEC network from different types of attacks and threats. A list of security and privacy challenges with their countermeasures is summarized in Table 3.

6 Security Attacks in MEC

The MEC provides a better structure for processing the data with low latency and low transmission delay. Due to the nature of distributed computing, several vulnerabilities and attacks are identified, which hamper the MEC network's security. Some of the attacks are discussed as follows:

- *Denial of Service Attacks*: In a DoS attack, the attackers' main aim is to disrupt the MEC services and block the resources provided by any applications. Distributed Denial of Service or DDoS attacks is the extension of DoS attacks in which attackers continuously send streams of the packet to the victim using distributed electronic devices, also known as botnets. This exhausts the hardware and resources of the victim. Resultant of this, the application is not able to process legitimate requests. Edge servers of the MEC network are more vulnerable to these attacks as they are computationally less powerful. Flood attacks are a type of DoS attack where the victim's system is flooded with malicious packets. UDP flooding [258, 259], ICMP flooding [260, 261], SYN flooding [262, 263], Ping of Death (PoD) [264] are some examples of flood attacks.
- *Zero-Day Attacks:* A zero-day attack is possible when an attacker determines the vulnerabilities in the software programs. The software developers and vendors are unaware of these unintentional flaws or holes in software programs. Attackers can exploit this undocumented vulnerability to achieve access to MEC servers and resources, which can lead to more issues [265, 266]. When the vendor discovers this issue, they begin to write a patch and tests it to resolve the software programs' weaknesses.
- *Poisoning attack:* Many types of AI applications use large datasets and intelligent algorithms to determine the patterns in the datasets. The software programs also learn the feature and patterns of the datasets. Based on these learning capabilities, they provide appropriate responses and predict future behavior. Attackers can provide the AI-based MEC system with wrong information, which decreases the AI model's accuracy. Manipulating data sets can also subtly change the design parameters to ignore suspicious activities [267, 268]. The poisoning attack includes data poisoning, model poisoning, and data modification.
- *Evasion attacks:* An evasion attack [269, 270] happens when the intruder is fed some perturbed input in the network. This input looks similar to the original one, and the original receiver does not identify the perturbed input. The inputs look the same to humans but throw the model off. For example, changing a few pixels in a photo will fail the image recognition system [271], but can seem normal to a human eye [272].
- *Exploiting Communication Channels:* A communication channel contains a lot of sensitive information about the communication entities. Thus, it is an attractive point for the attacker to exploit the communication network. These attacks may exploit packet streams or exploit wave signals [273, 274]. In this attack, an attacker continuously monitors network traffic to determine the communication entities sensitive information.
- *Malicious packet injections:* The attacker can inject the malicious packets either on the server-side or device side. SQL Injection [275], Cross-Site Scripting (XSS) [276, 277], Cross-Site Request Forgery (CSRF) [278] and Server-Side Request Forgery (SSRF) [279] are few attacks which target the edge servers. Bad actors try to inject the malicious packets into IoT-Edge servers to disturb the IoT

Table 3 Security challenges with their countermeasures

Challenge	Security Countermeasures adopted
Data poisoning	Auto encoder model diversity [216], gradient based optimization [217], Combining digital signatures and OTP based on hash chains [218], Data sanitization [219], and Anomaly detection [220]
Denial of services	Firewalls, Geo-blocking [221], Intrusion prevention, CAPTCHA [222], Account locking, Access control List [223], Filtering [224], and V-Guard prioritization integrated in VNF [225]
Data loss/data theft	Auditing methods [226], Data loss prevention system [227], Backup [228], OS baselining [229], and Anomaly detection [220]
Virtual machine sprawl	Policy implementation which automatically monitors and updates VM's [230], Storage optimization [231], and VM archiving
Virtual machine escape	Hypervisor patching [232], VM traffic monitoring [233], Administrative control [234], and VM segregation [235]
Weak authentication	Implement multi-factor authentication [236], Strong passwords [237], and light-weight-cryptographic algorithm [238, 239]
Data manipulation	Encrypted transmission [240], Integrity-checking [241], File integrity monitoring [242], and Logging activity [243]
Malicious code injection	Input validation [244], Parse tree validation [245], Query tokenization [246], and Context sensitive string evaluation [247]
Brute force	CAPTCHA [248], Account locking [223], Progressive delay, and Multi-factor authentication [236, 249]
Port scanning	Firewall [250], TCP wrapper [251], Intrusion prevention system [222], Using honeypots and honeynets [252]
Privacy preservation	Aggregation schemes such as homomorphic encryption [211], Monitor access logs, and Conduct employee security awareness training, Personal privacy evaluation, [253], and FL [131]
Data recovery	Implementing redundant array of inexpensive disks (RAID) [254], Manage access and control, and Backup at regular intervals
Data leakage	Hardware control, Data loss prevention System [255], Monitor access logs, Captcha Crypto Hash Functions [256], secure access control [257], and Encryption [183]

device functionality. Remotely injected malware can lead to remote code execution. These attacks can lead to data loss and theft, breach data integrity, and leak the password.

- *Dictionary attacks:* Authentication-based password protection mechanisms can protect the system from this attack [280, 281]. A dictionary attack is a type of brute force attack where all the possible combinations of passwords are tested to find the correct password. The difference between a brute force and dictionary attack is that dictionaries are commonly used passwords and are easily available and downloadable from open-source communities. This type of attack in the MEC server can degrade the strength of the authentication mechanism.
- *Weak authentication & Authorization Protocol:* Authentication is the process of uniquely identifying a person by using passwords. Weak passwords or weak authentication can lead the data loss or theft and identity theft. Bad actors target this weak authentication by using brute force or dictionary attacks. These vulnerabilities are also identified in WPA/WPA2 protocols [282, 283] in 4G and 5G networks. OAuth 1.0 [284, 285] authorization mechanism is vulnerable to fixation attacks [286, 287] in which service provider requests token is approved by some other relying party [288].
- *Sybil attack:* In Sybil attacks, the attacker show multiple pseudonymous identities to gain all the MEC privileges. This attack can lead to data loss or theft, breach the system's privacy, or hamper the reputation of the system [65].
- *Collusion attack:* It is a type of attack where an edge server or node has been compromised by making a secret agreement with bad actors [289]. There are two types of collusion—internal and external. In the MEC environment, the edge serves are automated, but it is possible with the collusion attack, the server is compromised. In this case, the server handling and maintenance activities are performed by a human.
- *Man in the Middle (MitM) attack:* In MitM, the attackers are come in between the communicating parties and impersonate the other party to receive data [150]. This can lead to data loss, privacy loss, replay attack, and data manipulation.

7 Applications of AI- and Fl-Based MEC

The MEC is considered a newly emerging technology. Due to its flexible and adaptable nature. It can be used in many applications where quick service response and high transmission rate are required. Some of the current notable applications of MEC can be summarized as follows:

7.1 Augmented Reality

This type of applications provides us with a reciprocal experience where the real-world environment is enhanced by computer-generated intuitive information. Augmented Reality uses information from the camera or location of the user, analyzes the data, and provides additional information about the things they are experiencing. The information is needed to be refreshed if there is movement. Once the data is generated, it is sent to the cloud server for analysis purposes, and the results are sent back to the user. However, this increases traffic a lot. MEC would answer this problem as information about the place would be stored locally on the node. This reduces the communication delay and also relieves the cloud server from additional traffic.

7.2 Internet of Things (IoT)

IoT devices generate lots of messages on telecom networks. A real-time capability and low-latency aggregation mechanisms are needed to handle the messages, protocols, message routing, and big data processing. MEC enables aggregation and distributes IoT services into base stations or Edge nodes which handles real-time responses. It also reduces the round-trip time of data.

7.3 Connected Cars

The architecture of connected cars is used to send vehicle-related data to the cloud servers so that users can get a better navigation system, reduce chances for road accidents, etc. MEC can join the connected car cloud with the MEC servers, enabling data and applications to be stored closer to cars. This would reduce the latency of data and provide real-time analysis.

7.4 Video Acceleration

Video acceleration applications can boost the QoE as well as improve resource utilization. All the web information can be accessed through Hypertext Transmission Protocol (HTTP) over the TCP protocol. In this situation, MEC can provide a better answer in terms of a fast response.

7.5 Smart City

To make the smart city, several wireless nodes, actuators, and sensors are placed in different locations in the city to measure the air, humidity, temperature, noise level, etc. All the sensors are interconnected with the help of the internet that provides a constant stream of information. This information can be analyzed to create a detailed report about the city and make technological improvements wherever necessary.

7.6 Smart Buildings

Smart buildings contain multiple MEC devices which can be placed at different levels in a building. It will also act as a Nano data center to collect information based on several factors. Emergency responses on fire can also be triggered using sensors. The MEC provides a solution to handle such emergency responses. Before allowing someone in the building, unique identification and authorization must be needed to improve the security of the building.

7.7 Connected Scenery Park

This type of application includes a network of MEC nodes to attend to local tourists' needs. The MEC nodes or servers should be installed in appropriate areas in the park. The nodes contain pre-loaded information with a map of the area and a tourist guide. They can also provide useful information about environmental monitoring, road conditions, and other safety information.

7.8 Big Data Analytics

MEC can provide a solution for handling big data problems. Latency is one of the important issues in the big data cloud. MEC processing units considered reliable data sources and proximity approach so that the latency issues cannot occur. A MEC server could handle the data accession, processing, minimize data movement, storage, and balancing computational abilities.

7.9 Blockchain

Integrating the current blockchain technology leverages computing power in the MEC environment. This makes incorporating more miners easier, which can increase the robustness of the blockchain-based mobile network. Additionally, the mobile users have an incentive from the reward obtained in the consensus process.

7.10 Smart Grid

The smart-grid-based applications contain multi-levelled architecture, including grid sensors and devices capable of handling time-sensitive and real-time data processing. These capabilities also extend machine-to-machine connectivity in which latency-sensitive data are processed very effectively.

7.11 Computational Offloading

This application transfer high computational tasks that need many resources to an external platform like an edge node, grid, or cloud. Offloading at the edge node reduces network communication load. It also reduces the access delay with respect to response time.

7.12 Content Delivery

Content delivery applications based on a content distribution network (CDN) are composed of proxy servers, edge nodes, and base stations geographically dispersed. The network goal is to provide services and data to its end users whenever needed in a high performance and efficient manner.

7.13 Collaborative Computing

Collaborative computing includes modern technological resources to promote and improve group work. It is a type of distributed technology where individuals work together from different locations. Combining MEC and 5G could result in a real-time context-aware ad-hoc collaboration framework. This framework effectively addresses the low latency cases as well as provides low-cost working platforms.

7.14 Healthcare

Edge computing technology has made telehealth and remote patient monitoring more accessible. This is possible because it takes low computation power, and the response is rapid. Internet of Medical Things (IoMT) [290] enables such medical systems in connected devices to give a fast response and take low time for the diagnosis of patients [291], as essential urgent service in some critical situations [292].

8 Discussion, Open Issues, and Future Scope

The prime objective of MEC is to deal with the issues present in the cloud computing model. Such issues are high bandwidth, high computation power, high service latency, and many more. The paper gives a brief introduction of the technologies currently used and points out the issues with the technology. The issues present in the current technology motivate us to investigate a new solution that overcomes such issues. This motivation leads to the invention of MEC, where the computation and processing units are near the user location. This will reduce the transmission distance as well as improve service efficiency. Several literature works are investigated to overcome the cloud computing issues by introducing the concept of MEC. Some of the works tried to improve the network efficiency of MEC. This work also considers the machine learning and deep learning-based AI approaches because they make the system more effective, faster, and intelligent.

The MEC background is thoroughly investigated, which will help other researchers understand the concept of MEC. This survey is not limited to MEC but also introduced the FL and IoT solutions used in MEC. Several works are presented in the literature to solve the Security and privacy threats and attacks. But, still, some of the threats and attacks are present. This works tried to provides some countermeasures to those threats and attacks. MEC can be used in many applications, including augmented reality, healthcare, IoT, Big data analytics, and many more.

MEC received much attention from the past few years, and people are more focused on this area. New researchers try to solve the problem present in the MEC. The literature identified several open issues that are still not solved and waiting for an effective solution. Some of the important open issues are as follows:

- Efficient Deployment and Management: MEC depends on the use of Edge nodes. All the MEC nodes need to be appropriately distributed in the network area. The properly distributed edge nodes give a guarantee of efficient MEC services for all users. Some algorithm and techniques need to be instigated that provides efficient deployment and management of MEC servers. This would make effective computational ability usage in terms of QoS and QoE. An efficient control procedure is also required to ensure proper management of MEC resources.
- Offloading Management: Offloading the jobs from the core network is one of the primary functions of MEC. It determines where the computation will be done—locally, edge node, or jointly. Most of the offloading research is based on hypothetical assumptions like users are not moving and focus mainly on power consumption. There is a lack of available research on the dynamic or moving user equipment. There is also a need for research on the effect of channel quality on offloading.
- Allocation of Computational Resources: In MEC, some techniques are needed for the efficient allocation of computational resources. During the process, if some resources are free, then immediately assign those resources to another process. This type of resource allocation system is missing in the literature.
- Standard Protocol: MEC is still in the developmental stage. There is a need to standardize the technology through a collaborative effort of different companies and researchers.
- Availability and Security: The resources should always be available to the user devices. This depends on the edge node capacity and the medium through which it is being accessed. The security of the data and applications from attackers should be considered shortly.
- Simulation Platform: Simulation platform creates a real-world system model using the programming language. This comes with many advantages, but some defects in the platform can create lots of software issues during the

developmental stage. Techniques of handling such issues are still open issues.

- Mobility Management: Continuously connection is one of the major problems while implementing a mobility management technique. The system needed a continuous connection with the edge server in both types of horizontal and vertical mobility.
- Pricing Model: A suitable pricing model should be established for consumers consuming edge networks from local or roaming-based stations. The pricing model should also keep the basic network parameters such as service cost, service response time, turnaround time, access bandwidth, and availability information in mind while creating the model.
- Transparent Application Migration: User applications send data to edge servers for execution. There needs a transparent migration system that migrates delay-sensitive and real-time applications to other platforms.
- Openness of Network: In the current architecture system, the network providers have complete control over the network. But, in MEC, the network is completely open in which different types of third-party vendors have come. They perform a different type of operation as well as access different data. This will increase the security risks in the system. A standard authentication mechanism and proper validation are needed before performing any action in the system.

9 Conclusion

Nowadays, MEC has emerged as novel technology integrated with other approaches to provide efficient practical services and decisions. Smart services and decisions take low computation power, fast response, required low bandwidth and offloading resource-intensive work. It also shifts the network architecture from centralized arrangement to decentralized effective architecture. This phenomenon improved the security of the architecture and protected the system from a single point failure. However, several challenges came to picture as obstacles to the deployment of MEC applications in different sectors. The challenges are not limited to security and privacy issues but also cover the creation of a standard deployment model, typical protocol, and offloading algorithms that are easily utilized in any IT system.

This paper tries to cover an updated review of such MEC challenges studying design standard platforms that are efficiently easily adaptable. The introduction of MEC provides standard architecture and deployment templates for the development of new applications. The FL and AI approaches involve the MEC system more flexible and smarter, such that they can be used attractively in many applications. Thus, the survey discussion also covers the new technological aspects like IoT and ML approaches used within MEC strategies. Due to the integration of innovative MEC technologies, several security and privacy challenges are coming vital into the system. Thus, the countermeasures of challenges found making all effort to overcome these tests issues. Due to the MEC dynamic and attractive nature, several attacks may hamper the network's functionality as covered differently within the paper presentation. The paper work highlights most related intellectual open issues and challenges that are still unsolved and need to be more focused on for future sophisticated research studies to come.

Acknowledgements Thanks for the international scientific collaboration of computer engineering scholars from institutions, KIIT Deemed to be University, Bhubaneswar-Odisha (India), and Umm Al-Qura University, Saudi Arabia. We appreciate motivating this wonderful survey work of artificial intelligence based mobile edge computing for internet of things research.

Funding There is no funding source.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

1. Kreibich, C.; Weaver, N.; Nechaev, B.; Paxson, V.: Netalyzr: illuminating the edge network. In: ACM SIGCOMM Conference on Internet Measurement, pp. 246–259 (2010)
2. Sun, X.; Ansari, N.: EdgeIoT: mobile edge computing for the Internet of Things. IEEE Commun. Mag. **54**(12), 22–29 (2016)
3. Fernando, N.; Loke, S.W.; Rahayu, W.: Mobile cloud computing: a survey. Fut Gener Comput Syst **29**(1), 84–106 (2013)
4. Gupta, P.; Gupta, S.: Mobile cloud computing: the future of cloud. Int. J. Adv. Res. Electr. Electron. Instrum. Eng. **1**(3), 134–145 (2012)
5. Qi, H.; Gani, A.: Research on mobile cloud computing: review, trend and perspectives. In: IEEE international conference on digital information and communication technology and its applications (DICTAP), pp. 195–202 (2012)
6. Harman, G.: Intrinsic qualities of experience. Philos. Perspect. **4**, 31–52 (1990)
7. Satyanarayanan, M.; Bahl, P.; Caceres, R.; Davies, N.: The case for VM-based cloudlets in mobile computing. IEEE Pervasive Comput. **8**(4), 14–23 (2009)
8. Patel, M.; Naughton, B.; Chan, C.; Sprecher, N.; Abeta, S.; Neal, A.; et al.: Mobile edge computing a key technology towards 5G. White paper, mobile-edge computing (MEC) industry initiative 29:854–864, 2014.
9. Lyytinen, K.; Yoo, Y.: Ubiquitous computing. Commun. ACM **45**(12), 63–96 (2002)

10. Huh, S.; Cho, S.; Kim, S.; Managing IoT devices using blockchain platform. In: IEEE International Conference on Advanced Communication Technology (ICACT), pp. 464–467 (2017)
11. Meneghello, F.; Calore, M.; Zucchetto, D.; Polese, M.; Zanella, A.: IoT: internet of threats? A survey of practical security vulnerabilities in real IoT devices. IEEE Internet Things J. **6**(5), 8182–8201 (2019)
12. Xiao, L.; Wan, X.; Xiaozhen, Lu.; Zhang, Y.; Di, Wu.: IoT security techniques based on machine learning: how do IoT devices use AI to enhance security? IEEE Signal Process. Mag. **35**(5), 41–49 (2018)
13. Floyd, S.; Jacobson, V.: Link-sharing and resource management models for packet networks. IEEE/ACM Trans. Netw. **3**(4), 365–386 (1995)
14. Foster, I.; Kesselman, C.; Lee, C.; Lindell, B.; Nahrstedt, K.; Roy, A.: A distributed resource management architecture that supports advance reservations and co-allocation. In: IEEE International Workshop on Quality of Service. IWQoS'99. (Cat. No. 98EX354), pp. 27–36 (1999)
15. Glasmann, J.; M¨uller, H.: Resource management architecture for realtime traffic in intranets. In: Networks, pp. 89–101. World Scientific (2002)
16. Zhang, Y.; Lan, X.; Li, Y.; Cai, L.; Pan, J.: Efficient computation resource management in mobile edge-cloud computing. IEEE Internet Things J. **6**(2), 3455–3466 (2018)
17. Qian, L.P.; Feng, A.; Huang, Y.; Wu, Y.; Ji, B.; Shi, Z.: Optimal SIC ordering and computation resource allocation in MEC-aware NOMA NB-IoT networks. IEEE Internet Things J **6**(2), 2806–2816 (2018)
18. Sanchez-Iborra, R.; Sanchez-Gomez, J.; Skarmeta, A.: Evolving IoT networks by the confluence of MEC and LP-WAN paradigms. Futur. Gener. Comput. Syst. **88**, 199–208 (2018)
19. Zhao, Z.; Zhao, R.; Xia, J.; Lei, X.; Li, D.; Yuen, C.; Fan, L.: A novel framework of three-hierarchical offloading optimization for MEC in industrial IoT networks. IEEE Trans. Ind. Inf. **16**(8), 5424–5434 (2019)
20. Ma, X.; Sun, H.; Hu, R.Q.: Scheduling policy and power allocation for federated learning in NOMA based MEC (2020). arXiv:2006.13044
21. Majeed, U.; Hong, C.S.; FLchain: federated learning via MEC-enabled blockchain network. In: IEEE Asia-Pacific Network Operations and Management Symposium (APNOMS), pp. 1–4 (2019)
22. Gutub, A.; Al-Qurashi, A.: Secure shares generation via M-blocks partitioning for counting-based secret sharing. J. Eng. Res. (JER) **8**(3), 91–117 (2020)
23. Chiang, M.; Zhang, T.: Fog and IoT: an overview of research opportunities. IEEE Internet Things J. **3**(6), 854–864 (2016)
24. Burdea, G.C.; Coiffet, P.: Virtual Reality Technology. Wiley, Hoboken (2003)
25. Nguyen, M.; Tran, N.; Tun, Y.; Han, Z.; Hong, C.: Toward multiple federated learning services resource sharing in mobile edge networks. IEEE Trans. Mob. Comput. (2021). https://doi.org/10.1109/TMC.2021.3085979
26. Basta, A.; Kellerer, W.; Hoffmann, M.; Morper, H.J.; Hoffmann, K.: Applying NFV and SDN to LTE mobile core gateways, the functions placement problem. In: AllThingsCellular'14—workshop on All things cellular: operations, applications, & challenges, pp. 33–38 (2014)
27. Hawilo, H.; Shami, A.; Mirahmadi, M.; Asal, R.: NFV: state of the art, challenges, and implementation in next generation mobile networks (vEPC). IEEE Network **28**(6), 18–26 (2014)
28. Matias, J.; Garay, J.; Toledo, N.; Unzilla, J.; Jacob, E.: Toward an SDN-enabled NFV architecture. IEEE Commun. Mag. **53**(4), 187–193 (2015)
29. Fayazbakhsh, S.K.; Lin, Y.; Tootoonchian, A.; Ghodsi, A.; Koponen, T.; Maggs, B.; Ng, K.C.; Sekar, V.; Shenker, S.: Less pain, most of the gain: incrementally deployable ICN. In: ACM SIGCOMM Computer Communication Review, 43(4):147–158 (2013)
30. Ion, M.; Zhang, J.; Schooler, E.M.: Toward content-centric privacy in ICN: attribute-based encryption and routing. In: ACM SIGCOMM workshop on Information-centric networking, pp. 39–40 (2013)
31. Ravindran, R.; Chakraborti, A.; Amin, S.O.; Azgin, A.; Wang, G.: 5G-ICN: delivering ICN services over 5G using network slicing. IEEE Commun. Mag. **55**(5), 101–107 (2017)
32. Altalhi, S.; Gutub, A.: A survey on predictions of cyber-attacks utilizing real-time twitter tracing recognition. J. Ambient Intell. Hum. Comput. (2021). https://doi.org/10.1007/s12652-020-02789-z
33. Abbas, N.; Zhang, Y.; Taherkordi, A.; Skeie, T.: Mobile edge computing: a survey. IEEE Internet Things J. **5**(1), 450–465 (2017)
34. Mao, Y.; You, C.; Zhang, J.; Huang, K.; Letaief, K.B.: A survey on mobile edge computing: the communication perspective. IEEE Commun. Surv. Tutor. **19**(4), 2322–2358 (2017)
35. Mach, P.; Becvar, Z.: Mobile edge computing: a survey on architecture and computation offloading. IEEE Commun. Surv. Tutor. **19**(3), 1628–1656 (2017)
36. Hibat Allah, B.; Abdellah, I.: MEC towards 5G: A survey of concepts, use cases, location tradeoffs. Trans. Mach. Learn. Artif. Intell. (2017). https://doi.org/10.14738/tmlai.54.3215
37. Taleb, T.; Samdanis, K.; Mada, B.; Flinck, H.; Dutta, S.; Sabella, D.: On multi-access edge computing: a survey of the emerging 5G network edge cloud architecture and orchestration. IEEE Commun. Surv. Tutor. **19**(3), 1657–1681 (2017)
38. Porambage, P.; Okwuibe, J.; Liyanage, M.; Ylianttila, M.; Taleb, T.: Survey on multi-access edge computing for internet of things realization. IEEE Commun. Surv. Tutor. **20**(4), 2961–2991 (2018)
39. Moura, J.; Hutchison, D.: Game theory for multi-access edge computing: survey, use cases, and future trends. IEEE Commun. Surv. Tutor. **21**(1), 260–288 (2018)
40. Yousefpour, A.; Fung, C.; Nguyen, T.; Kadiyala, K.; Jalali, F.; Niakanlahiji, A.; Kong, J.; Jue, J.P.: All one needs to know about fog computing and related edge computing paradigms: a complete survey. J. Syst. Archit. **98**, 289–330 (2019)
41. Mehrabi, M.; You, D.; Latzko, V.; Salah, H.; Reisslein, M.; Fitzek, F.H.P.: Device-enhanced MEC: multi-access edge computing (MEC) aided by end device computation and caching: a survey. IEEE Access **7**, 166079–166108 (2019)
42. Li, Q.; Wen, Z.; Wu, Z.; Hu, S.; Wang, N.; He, B.: A survey on federated learning systems: vision, hype and reality for data privacy and protection (2019). arXiv:1907.09693
43. Aledhari, M.; Razzak, R.; Parizi, R.M.; Saeed, F.: Federated learning: a survey on enabling technologies, protocols, and applications. IEEE Access **8**, 140699–140725 (2020)
44. Pham, Q.-V.; Fang, F.; Ha, V.N.; Piran, M.J.; Le, M.; Le, L.B.; Hwang, W.-J.; Ding, Z.: A survey of multi-access edge computing in 5G and beyond: fundamentals, technology integration, and state-of-the-art. IEEE Access **8**, 116974–117017 (2020)
45. Lim, W.Y.B.; Luong, N.C.; Hoang, D.T.; Jiao, Y.; Liang, Y.-C.; Yang, Q.; Niyato, D.; Miao, C.: Federated learning in mobile edge networks: a comprehensive survey. IEEE Commun. Surv. Tutor. **22**(3), 2031–2063 (2020)
46. Spinelli, F.; Mancuso, V.: Toward enabled industrial verticals in 5G: a survey on MEC-based approaches to provisioning and flexibility. IEEE Commun. Surv. Tutor. **23**(1), 596–630 (2021)
47. Chamikara, M.A.P.; Bertok, P.; Khalil, I.; Liu, D.; Camtepe, S.: Privacy preserving distributed machine learning with federated learning. Comput. Commun. **171**, 112–125 (2021)

48. Feng, C.; Zhao, Z.; Wang, Y.; Quek, T.Q.S.; Peng, M.: On the design of federated learning in the mobile edge computing systems. IEEE Trans. Commun. **69**(9), 5902–5916 (2021)
49. Guo, Y.; Zhao, Z.; He, K.; Lai, S.; Xia, J.; Fan, L.: Efficient and flexible management for industrial internet of things: a federated learning approach. Comput. Netw. **192**, 108122 (2021)
50. Li, X.; Cheng, L.; Sun, C.; Lam, K.-Y.; Wang, X.; Li, F.: Federated-learning-empowered collaborative data sharing for vehicular edge networks. IEEE Netw. **35**(3), 116–124 (2021)
51. Lu, R.; Zhang, W.; Li, Q.; Zhong, X.; Vasilakos, A.V: Auction based clustered federated learning in mobile edge computing system (2021). arXiv:2103.07150
52. Makkar, A.; Ghosh, U.; Rawat, D.B.; Abawajy, J.: FedLearnSP: preserving privacy and security using federated learning and edge computing. IEEE Consum. Electron. Mag. (2021). https://doi.org/10.1109/MCE.2020.3048926
53. Nguyen, D.C.; Ding, M.; Pham, Q.-V.; Pathirana, P.N.; Le, L.B.; Seneviratne, A.; Li, J.; Niyato, D.; Poor, H.V.: Federated learning meets blockchain in edge computing: opportunities and challenges. IEEE Internet Things J. **8**(16), 12806–12825 (2021)
54. Yan, H.; Li, Hu.; Xiang, X.; Liu, Z.; Yuan, Xu.: Ppcl: Privacy-preserving collaborative learning for mitigating indirect information leakage. Inf. Sci. **548**, 423–437 (2021)
55. Zhang, P.; Wang, C.; Jiang, C.; Han, Z.: Deep reinforcement learning assisted federated learning algorithm for data management of iiot. IEEE Trans. Industr. Inf. **17**(12), 8475–8484 (2021)
56. Lillicrap, T.P.; Hunt, J.J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D.: Continuous control with deep reinforcement learning (2019). arXiv:1509.02971
57. Abdelwahab, S.; Hamdaoui, B.; Guizani, M.; Znati, T.: Network function virtualization in 5G. IEEE Commun. Mag. **54**(4), 84–91 (2016)
58. He, Y.; Liang, C.; Yu, F.R.; Zhao, N.; Yin, H.: Optimization of cache-enabled opportunistic interference alignment wireless networks: a big data deep reinforcement learning approach. In: IEEE International Conference on Communications (ICC), pp. 1–6 (2017)
59. Guo, K.; Yang, C.; Liu, T.: Caching in base station with recommendation via Q-learning. In: IEEE Wireless Communications and Networking Conference (WCNC), pp. 1–6 (2017)
60. He, X.; Liu, J.; Jin, R.; Dai, H.: Privacy-aware offloading in mobile-edge computing. In: GLOBECOM IEEE Global Communications Conference, pp. 1–6, 2017.
61. Huang, X.; Yuan, T.; Qiao, G.; Ren, Y.: Deep reinforcement learning for multimedia traffic control in software defined networking. IEEE Netw. **32**(6), 35–41 (2018)
62. Zhang, N.; Zheng, K.; Tao, M.: Using grouped linear prediction and accelerated reinforcement learning for online content caching. In: IEEE International Conference on Communications Workshops (ICC Workshops), pp. 1–6 (2018)
63. Bhagoji, A.N.; Chakraborty, S.; Mittal, P.; Calo, S.: Analyzing federated learning through an adversarial lens. In: PMLR International Conference on Machine Learning, pp. 634–643 (2019)
64. Bhowmick, A.; Duchi, J.; Freudiger, J.; Kapoor, G.; Rogers, R.: Protection against reconstruction and its applications in private federated learning (2018). arXiv:1812.00984
65. Clement, F.; Chris, J.M.Y.; Ivan, B.: Mitigating sybils in federated learning poisoning (2018). arXiv:1808.04866
66. Ma, L.; Liu, X.; Pei, Q.; Xiang, Y.: Privacy-preserving reputation management for edge computing enhanced mobile crowdsensing. IEEE Trans. Serv. Comput. **12**(5), 786–799 (2018)
67. Zhang, X.D.; Li, R.; Cui, B.: A security architecture of VANET based on blockchain and mobile edge computing. In: IEEE International Conference on Hot Information-Centric Networking (HotICN), pp. 258–259 (2018)
68. Syamkumar, M.; Barford, P.; Durairajan, R.: Deployment characteristics of "the edge" in mobile edge computing. In: Proceedings of the 2018 workshop on mobile edge communications, pp. 43–49 (2018)
69. Li, C.-Y.; Liu, H.-Y.; Huang, P.-H.; Chien, H.-T.; Tu, G.-H.; Hong, P.-Y.; Lin, Y.-D.: Mobile edge computing platform deployment in 4G LTE networks: a middlebox approach. In: fUSENIXg Workshop on Hot Topics in Edge Computing (HotEdge 18) (2018)
70. Milad Nasr, Reza Shokri, and Amir Houmansadr. Comprehensive Privacy Analysis of Deep Learning: Passive and Active White-box Inference Attacks against Centralized and Federated Learning. IEEE symposium on security and privacy (SP), pages 739–753, 2019.
71. Xiaolong, Xu.; Zhang, X.; Gao, H.; Xue, Y.; Qi, L.; Dou, W.: BeCome: blockchain-enabled computation offloading for IoT in mobile edge computing. IEEE Trans. Industr. Inf. **16**(6), 4187–4195 (2019)
72. Gai, K.; Yulu, Wu.; Zhu, L.; Lei, Xu.; Zhang, Y.: Permissioned blockchain and edge computing empowered privacy-preserving smart grid networks. IEEE Internet Things J. **6**(5), 7992–8004 (2019)
73. Xiaolong, Xu.; He, C.; Zhanyang, Xu.; Qi, L.; Wan, S.; Bhuiyan, M.Z.A.: Joint optimization of offloading utility and privacy for edge computing enabled IoT. IEEE Internet Things J. **7**(4), 2622–2629 (2019)
74. Pang, M.; Wang, Li.; Fang, N.: A collaborative scheduling strategy for IoV computing resources considering location privacy protection in mobile edge computing environment. J. Cloud Comput. **9**(1), 1–17 (2020)
75. Zhanyang, Xu.; Liu, X.; Jiang, G.; Tang, B.: A time-efficient data offloading method with privacy preservation for intelligent sensors in edge computing. EURASIP J. Wirel. Commun. Netw. **2019**(1), 1–12 (2019)
76. Hui, H.; Zhou, C.; An, X.; Lin, F.: A new resource allocation mechanism for security of mobile edge computing system. IEEE Access **7**, 116886–116899 (2019)
77. Gyamfi, E.; Ansere, J.A.; Xu, L.: ECC based lightweight cybersecurity solution for IoT networks utilising multi-access mobile edge computing. In: IEEE International Conference on Fog and Mobile Edge Computing (FMEC), pp. 149–154 (2019)
78. Huang, B.; Li, Z.; Tang, P.; Wang, S.; Zhao, J.; Haiyang, Hu.; Li, W.; Chang, V.: Security modeling and efficient computation offloading for service workflow in mobile edge computing. Futur. Gener. Comput. Syst. **97**, 755–774 (2019)
79. Ranaweera, P.; Jurcut, A.D.; Liyanage, M.: Realizing multi-access edge computing feasibility: security perspective. In: IEEE Conference on Standards for Communications and Networking (CSCN), pp. 1–7 (2019)
80. Truex, S.; Baracaldo, N.; Anwar, A.; Steinke, T.; Ludwig, H.; Zhang, R.; Zhou, Y.: A hybrid approach to privacy-preserving federated learning. ACM Workshop on Artificial Intelligence and Security, pp. 1–11 (2019)
81. Yunlong, Lu.; Huang, X.; Dai, Y.; Maharjan, S.; Zhang, Y.: Federated learning for data privacy preservation in vehicular cyber-physical systems. IEEE Netw. **34**(3), 50–56 (2020)
82. Bagdasaryan, E.; Veit, A.; Hua, Y.; Estrin, D.; Shmatikov, V.: How to backdoor federated learning. In: PMLR International Conference on Artificial Intelligence and Statistics, pp. 2938–2948 (2020)
83. Rong, Yu.; Li, P.: Toward resource-efficient federated learning in mobile edge computing. IEEE Netw. **35**(1), 148–155 (2021)
84. Liu, Y.; Youyang, Qu.; Chenhao, Xu.; Hao, Z.; Bruce, Gu.: Blockchain-enabled asynchronous federated learning in edge computing. Sensors **21**(10), 3335 (2021)
85. Shahidinejad, A.; Farahbakhsh, F.; Ghobaei-Arani, M.; Malik, M.H.; Anwar, T.: Context-aware multi-user offloading in mobile

edge computing: a federated learning-based approach. J. Grid Comput. **19**(2), 1–23 (2021)
86. Almutairi, S.; Gutub, A.; Al-Juaid, N.: Motivating teachers to use information technology in educational process within Saudi Arabia. Int. J. Technol. Enhanc. Learn. (IJTEL) **12**(2), 200–217 (2020)
87. Zhang, J.; Zhao, Y.; Wang, J.; Chen, B.: FedMEC: improving efficiency of differentially private federated learning via mobile edge computing. Mobile Netw. Appl. **25**(6), 2421–2433 (2020)
88. He, X.; Jin, R.; Dai, H.: Physical-layer assisted privacy-preserving offloading in mobile-edge computing. In: ICC IEEE International Conference on Communications (ICC), pp. 1–6, 2019
89. Yang, H.; Liang, Y.; Yuan, J.; Yao, Q.; Ao, Yu.; Zhang, J.: Distributed blockchain-based trusted multidomain collaboration for mobile edge computing in 5G and beyond. IEEE Trans. Industr. Inf. **16**(11), 7094–7104 (2020)
90. He, X.; Jin, R.; Dai, H.: Peace: privacy-preserving and cost-efficient task offloading for mobile-edge computing. IEEE Trans. Wirel. Commun. **19**(3), 1814–1824 (2019)
91. Porambage, P.; Kumar, Y.; Liyanage, M.; Partala, J.; Lov´en, L.; Ylianttila, M.; Sepp¨anen, T.: Sec-EdgeAI: AI for edge security vs security for edge AI. The 1st 6G Wireless Summit,(Levi, Finland) (2019)
92. Feibo, J.; Kezhi, W.; Li, D.; Cunhua, P.; Wei, X.; Kun, Y.: AI driven heterogeneous MEC system with UAV assistance for dynamic environment: challenges and solutions. IEEE Network (2020)
93. Benzaid, C.; Taleb, T.: AI for beyond 5G networks: a cyber-security defense or offense enabler? IEEE Netw. **34**(6), 140–147 (2020)
94. Lin, J.; Wei, Yu.; Zhang, N.; Yang, X.; Zhang, H.; Zhao, W.: A survey on internet of things: architecture, enabling technologies, security and privacy, and applications. IEEE Internet Things J. **4**(5), 1125–1142 (2017)
95. Zhang, D.; Ma, Y.; Hu, X.S.; Wang, D.: Toward privacy-aware task allocation in social sensing-based edge computing systems. IEEE Internet Things J. **7**(12), 11384–11400 (2020)
96. Gheisari, M.; Pham, Q.-V.; Alazab, M.; Zhang, X.; Fernandez-Campusano, C.; Srivastava, G.: ECA: an edge computing architecture for privacy-preserving in IoT-based smart city. IEEE Access **7**, 155779–155786 (2019)
97. Vance, N.; Zhang, D.; Zhang, Y.; Wang, D.: Privacy-aware edge computing in social sensing applications using ring signatures. In: IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS), pp. 755–762 (2018).
98. Alwarafy, A.; Al-Thelaya, K.A.; Abdallah, M.; Schneider, J.; Hamdi, M.: A survey on security and privacy issues in edge-computing-assisted internet of things. IEEE Internet Things J. **8**(6), 4004–4022 (2020)
99. Mao, Y.; You, C.; Zhang, J.; Huang, K.; Letaief, K.B.: Mobile edge computing: survey and research outlook (2017). arXiv:1701.01090v3
100. Wang, S.; Zhang, X.; Zhang, Y.; Wang, L.; Yang, J.; Wang, W.: A survey on mobile edge networks: convergence of computing. Caching Commun. IEEE Access **5**, 6757–6779 (2017)
101. Farooqi, N.; Gutub, A.; Khozium, M.: Smart community challenges: enabling IoT/M2M technology case study. Life Sci. J. **16**(7), 11–17 (2019)
102. Xiao, L.; Wan, X.; Dai, C.; Xiaojiang, Du.; Chen, X.; Guizani, M.: Security in mobile edge caching with reinforcement learning. IEEE Wirel. Commun. **25**(3), 116–122 (2018)
103. Gutub, A.: Regulating watermarking semi-authentication of multimedia audio via counting-based secret sharing. Pamukkale Univ J. Eng. Sci (2021). https://doi.org/10.5505/pajes.2021.54837
104. Mtibaa, A.; Harras, K.; Alnuweiri, H.: Friend or foe? Detecting and isolating malicious nodes in mobile edge computing platforms. In: IEEE International Conference on Cloud Computing Technology and Science (CloudCom), pp. 42–49 (2015)
105. Almajali, S.; Salameh, H.B.; Ayyash, M.; Elgala, H.: A framework for efficient and secured mobility of IoT devices in mobile edge computing. In: IEEE International Conference on Fog and Mobile Edge Computing (FMEC 2018), pp. 58–62 (2018)
106. Rathore, S.; Sharma, P.K.; Sangaiah, A.K.; Park, J.J.: A hesitant fuzzy based security approach for fog and mobile-edge computing. IEEE Access **6**, 688–701 (2017)
107. Jorge, M.-P.; Cominardi, L.; Bernardos, C.J.; de la Oliva, A.; Azcorra, A.: Modeling mobile edge computing deployments for low latency multimedia services. IEEE Trans. Broadcast. **65**(2), 464–474 (2019)
108. Gutub, A.: Efficient utilization of scalable multipliers in parallel to compute GF(p) elliptic curve cryptographic operations. Kuwait J. Sci. Eng. (KJSE) **34**(2), 165–182 (2007)
109. Jia, X.; He, D.; Kumar, N.; Choo, K.-K.R.: A provably secure and efficient identity-based anonymous authentication scheme for mobile edge computing. IEEE Syst. J. **14**(1), 560–571 (2019)
110. Zhou, Y.; Pan, C.; Yeoh, P.L.; Wang, K.; Elkashlan, M.; Vucetic, B.; Li, Y.: Secure communications for UAV-enabled mobile edge computing systems. IEEE Trans. Commun. **68**(1), 376–388 (2019)
111. He, D.; Chan, S.; Guizani, M.: Security in the internet of things supported by mobile edge computing. IEEE Commun. Mag. **56**(8), 56–61 (2018)
112. Al-Zinati, M.; Almasri, T.; Alsmirat, M.; Jararweh, Y.: Enabling multiple health security threats detection using mobile edge computing. Simul. Modell. Pract. Theory **101**, 101957 (2020)
113. Nilsson, A.; Smith, S.; Ulm, G.; Gustavsson, E.; Jirstrand, M.: A performance evaluation of federated learning algorithms. Second Workshop on Distributed Infrastructures for Deep Learning, pp. 1–8 (2018)
114. Ahmad, I.; Kumar, T.; Liyanage, M.; Okwuibe, J.; Ylianttila, M.; Gurtov, A.: Overview of 5G security challenges and solutions. IEEE Commun. Stand. Mag. **2**(1), 36–43 (2018)
115. Kewei, C.; Tao, F.; Yilun, J.; Yang, L.; Tianjian, C.; Qiang, Y.: SecureBoost: a lossless federated learning framework (2021). arXiv:1901.08755
116. Al-Shaarani, F.; Gutub, A.: Increasing participants using counting-based secret sharing via involving matrices and practical steganography. Arab. J. Sci. Eng. (AJSE) **1**, 2 (2021). https://doi.org/10.1007/s13369-021-06165-7
117. Bissmeyer, N.; van Dam, J.-F.; Zimmermann, C.; Eckert, K.; Security in hybrid vehicular communication based on ITS-G5, LTE-V, and mobile edge computing. In: AmE 2018-automotive meets electronics; 9th GMM-Symposium, pp. 1–6. VDE (2018)
118. Hou, Y.; Garg, S.; Hui, L.; Nalin, D.; Jayakody, K.; RJin, M S Hossain,: A data security enhanced access control mechanism in mobile edge computing. IEEE Access **8**, 136119–136130 (2020)
119. Belli, D.; Chessa, S.; Foschini, L.; Girolami, M.: A probabilistic model for the deployment of human-enabled edge computing in massive sensing scenarios. IEEE Internet Things J. **7**(3), 2421–2431 (2019)
120. Mohri, M.; Sivek, G.; Suresh, A.T.: Agnostic federated learning. In: PMLR International Conference on Machine Learning, pp. 4615–4625 (2019)
121. Wang, En.; Li, D.; Dong, B.; Zhou, H.; Zhu, M.: Flat and hierarchical system deployment for edge computing systems. Futur. Gener. Comput. Syst. **105**, 308–317 (2020)
122. Elgendy, I.A.; Zhang, W.; Tian, Y.-C.; Li, K.: Resource allocation and computation offloading with data security for mobile edge computing. Fut. Gener. Comput. Syst. **100**, 531–541 (2019)
123. Jere, S.; Fan, Q.; Shang, B.; Li, L.; Liu, L.: Federated learning in mobile edge computing: an edge-learning perspective for beyond 5G (2020). arXiv:2007.08030

124. Mohammad, U.; Sorour, S.: Adaptive task allocation for mobile edge learning. In: IEEE Wireless Communications and Networking Conference Workshop (WCNCW), pp. 1–6 (2019)
125. Wang, X.; Han, Y.; Wang, C.; Zhao, Q.; Chen, X.; Chen, M.: In-edge AI: intelligentizing mobile edge computing, caching and communication by federated learning. IEEE Netw. **33**(5), 156–165 (2019)
126. Chen, D.; Xie, L.J.; Kim, B.G.; Wang, L.; Hong, C.S.; Wang, L.-C.; Han, Z.: Federated learning based mobile edge computing for augmented reality applications. In: IEEE International Conference on Computing, Networking and Communications (ICNC), pp. 767–773 (2020)
127. Ho-Phuoc, T.: CIFAR10 to compare visual recognition performance between deep neural networks and humans (2018). arXiv:1811.07270
128. Li, H.; Liu, H.; Ji, X.; Li, G.; Shi, L.: CIFAR10-DVS: an event-stream dataset for object classification. Front. Neurosci. **11**, 309 (2017)
129. Recht, B.; Roelofs, R.; Schmidt, L.; Shankar, V.: Do CIFAR-10 classifiers generalize to CIFAR-10? (2018). arXiv:1806.00451
130. Kourtellis, N.; Katevas, K.; FLaaS, D.P.: Federated learning as a service. In: Proceedings of the 1st workshop on distributed machine learning, pp. 7–13 (2020)
131. Liu, G.; Wang, C.; Ma, X.; Yang, Y.: Keep your data locally: Federated-learning-based data privacy preservation in edge computing. IEEE Netw. **35**(2), 60–66 (2021)
132. Chen, N.; Li, Y.; Liu, X.; Zhang, Z.: A mutual information based federated learning framework for edge computing networks. Comput. Commun. **176**, 23–30 (2021)
133. Wang, Q.; Li, Q.; Wang, K.; Wang, H.; Zeng, P.: Efficient federated learning for fault diagnosis in industrial cloud-edge computing. Computing **103**(11), 2319–2337 (2021)
134. Li, H.; Shou, G.; Hu, Y.; Guo, Z.: Mobile edge computing: progress and challenges. In: IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (Mobile-Cloud), pp. 83–84 (2016)
135. Gerla, M.; Tsai, J.T.-C.: Multicluster, mobile, multimedia radio network. Wirel. Netw. **1**(3), 255–265 (1995)
136. Laiho, J.; Wacker, A.; Novosad, T.: Radio network planning and optimisation for UMTS, Vol. 2. Wiley, Hoboken (2002)
137. Giust, F.; Verin, G.; Antevski, K.; Chou, J.; Fang, Y.; Featherstone, W.; Fontes, F.; Frydman, D.; Li, A.; Manzalini, A.; et al.: MEC deployments in 4G and evolution towards 5G. ETSI White paper **24**(2018), 1–24 (2018)
138. Portnoy, M.: Virtualization Essentials, Vol. 19. Wiley, Hoboken (2012)
139. Uhlig, R.; Neiger, G.; Rodgers, D.; Santoni, A.L.; Martins, F.C.M.; Anderson, A.V.; Bennett, S.M.; Kagi, A.; Leung, F.H.; Smith, L.: Intel virtualization technology. Computer **38**(5), 48–56 (2005)
140. Ahmavaara, K.; Haverinen, H.; Pichna, R.: Interworking architecture between 3GPP and WLAN systems. IEEE Commun. Mag. **41**(11), 74–81 (2003)
141. Eric Wang, Y.-P.; Lin, X.; Adhikary, A.; Grovlen, A.; Sui, Y.; Blankenship, Y.; Bergman, J.; Razaghi, H.S.: A primer on 3GPP narrowband Internet of Things. IEEE Commun Mag **55**(3), 117–123 (2017)
142. Aly, S.; AlGhamdi, T.; Salim, M.; Amin, H.; Gutub, A.: Information gathering schemes for collaborative sensor devices. Procedia Comput. Sci. **32**, 1141–1146 (2014)
143. Hadzialic, M.; Dosenovic, B.; Dzaferagic, M.; Musovic, J.: Cloud-RAN: innovative radio access network architecture. In: IEEE Proceedings ELMAR, pp. 115–120 (2013)
144. Wu, J.; Zhang, Z.; Hong, Y.; Wen, Y.: Cloud radio access network (C-RAN): a primer. IEEE Netw. **29**(1), 35–41 (2015)
145. Alharthi, N.; Gutub, A.: Data visualization to explore improving decision-making within Hajj services. Sci. Modell. Res. **2**(1), 9–18 (2017)
146. Aly, S.; Alghamdi, T.; Salim, M.; Gutub, A.: Data dissemination and collection algorithms for collaborative sensor devices using dynamic cluster heads. Trends Appl. Sci. **8**(2), 55–72 (2013). https://doi.org/10.3923/tasr.2013.55.72
147. Ha, K.; Satyanarayanan, M.: Openstack++ for cloudlet deployment. School of Computer Science Carnegie Mellon University, Pittsburgh, CMU-CS-15-123 (2015)
148. Verbelen, T.; Simoens, P.; De Turck, F.; Dhoedt, B.: Cloudlets: bringing the cloud to the mobile user. ACM Workshop on Mobile Cloud Computing and Services, pp. 29–36 (2012)
149. Bonomi, F.; Milito, R.; Zhu, J.; Addepalli, S.: Fog computing and its role in the internet of things. In: Proceedings of the first edition of the MCC workshop on Mobile cloud computing, pp. 13–16 (2012)
150. Stojmenovic, I.; Wen, S.: The fog computing paradigm: scenarios and security issues. In: IEEE Federated Conference on Computer Science and Information Systems, pp. 1–8 (2014)
151. Yi, S.; Hao, Z.; Qin, Z.; Li, Q.: Fog computing: platform and applications. In 2015 Third IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb), pp. 73–78. IEEE (2015)
152. Klamt, S.; von Kamp, A.: An application programming interface for Cell NetAnalyzer. Biosystems **105**(2), 162–168 (2011)
153. Capozzi, F.; Piro, G.; Grieco, L.A.; Boggia, G.; Camarda, P.: Downlink packet scheduling in LTE cellular networks: key design issues and a survey. IEEE Commun. Surv. Tutor. **15**(2), 678–700 (2012)
154. Sadiq, B.; Madan, R.; Sampath, A.: Downlink scheduling for multiclass traffic in LTE. EURASIP J. Wirel. Commun. Netw. **1–18**, 2009 (2009)
155. Altay, C.; Bozdemir, N.Z.; Camcıo˘glu, E.: Standalone eNode-B design with integrated virtual EPC in public safety networks. In: NOMS IEEE/IFIP Network Operations and Management Symposium, pp. 731–734 (2016)
156. Ferng, H.-W.; Huang, Y.-Y.: Handover scheme with enode-B pre-selection and parameter self-optimization for LTE-A heterogeneous networks. IEEE Int. Conf. Mach. Learn. Cybern. (ICMLC) **2**, 594–599 (2016)
157. Dahlman, E.; Parkvall, S.; Skold, J.; Beming, P.: 3G evolution: HSPA and LTE for mobile broadband. Academic Press, Cambridge (2010)
158. Dahlman, E.; Parkvall, S.; Skold, J.: 4G: LTE/LTE-advanced for mobile broadband. Academic Press, Cambridge (2013)
159. Mijumbi, R.; Serrat, J.; Gorricho, J.-L.; Bouten, N.; De Turck, F.; Boutaba, R.: Network function virtualization: State-of-the-art and research challenges. IEEE Commun. Surv. Tutor. **18**(1), 236–262 (2015)
160. Gelberger, A.; Yemini, N.; Giladi, R.: Performance analysis of software-defined networking (SDN). In: IEEE International Symposium on Modelling, Analysis and Simulation of Computer and Telecommunication Systems, pp: 389–393 (2013)
161. Haleplidis, E.; Pentikousis, K.; Denazis, S.; Salim, J.H.; Meyer, D.; Koufopavlou, O.: Software-defined networking (SDN): layers and architecture terminology. RFC 7426 (2015)
162. MECISG ETSI. Mobile Edge Computing (MEC); Framework and Reference Architecture. ETSI, DGS MEC, 3 (2016)
163. Ahokangas, P.; Matinmikko, M.; Yrjola, S.; Okkonen, H.; Casey, T.: Simple rules" for mobile network operators' strategic choices in future cognitive spectrum sharing networks. IEEE Wirel. Commun. **20**(2), 20–26 (2013)
164. Banerjee, A.; Dippon, C.M.: Voluntary relationships among mobile network operators and mobile virtual network operators: an economic explanation. Inf. Econ. Policy **21**(1), 72–84 (2009)

165. Dewire, D.T.: Application service providers. Inf. Syst. Manag. **17**(4), 14–19 (2000)
166. Kakabadse, A.; Kakabadse, N.: Application service providers (ASPs): new impetus for transformational change. Knowl. Process Manag. **9**(4), 205–218 (2002)
167. Sharma, S.K.; Gupta, J.N.D.: Application service providers: issues and challenges. Logist. Inf. Manag. **15**(3), 160–169 (2002)
168. Beck, M.T.; Werner, M.; Feld, S.; Schimper, S.: Mobile edge computing: a taxonomy. In: Citeseer International Conference on Advances in Future Internet, pp. 48–55 (2014)
169. Bhardwaj, S.; Jain, L.; Jain, S.: Cloud computing: a study of infrastructure as a service (IAAS). Int. J. Eng. Inf. Technol. **2**(1), 60–63 (2010)
170. Malawski, M.; Juve, G.; Deelman, E.; Nabrzyski, J.: Algorithms for cost-and deadline-constrained provisioning for scientific workflow ensembles in IaaS clouds. Futur. Gener. Comput. Syst. **48**, 1–18 (2015)
171. Manvi, S.S.; Shyam, G.K.: Resource management for Infrastructure as a Service (IaaS) in cloud computing: a survey. J. Netw. Comput. Appl. **41**, 424–440 (2014)
172. Al-Shaarani, F.; Gutub, A.: Securing matrix counting-based secret-sharing involving crypto steganography. J. King. Saud Univ. Comput. Inf. Sci. (2021). https://doi.org/10.1016/j.jksuci.2021.09.009
173. TensorFlow Federated. Machine Learning on Decentralized Data. TensorFlow. URL: https://www.tensorflow.org/federated Accessed 13 Oct 2020 (2019)
174. Kholod, I.; Yanaki, E.; Fomichev, D.; Shalugin, E.; Novikova, E.; Filippov, E.; Nordlund, M.: Open-source federated learning frameworks for IoT: a comparative review and analysis. Sensors **21**(1), 167 (2021)
175. Rieke, N.; Hancox, J.; Li, W.; Milletari, F.; Roth, H.R.; Albarqouni, S.; Bakas, S.; Galtier, M.N.; Landman, B.A.; Maier-Hein, K.; et al.: The future of digital health with federated learning. NPJ Digit. Med. **3**(1), 1–7 (2020)
176. Tian, Z.; Zhang, R.; Hou, X.; Liu, J.; Ren, K.: FederBoost: private federated learning for GBDT (2020). arXiv:2011.02796
177. Xu, J.; Glicksberg, B.S.; Su, C.; Walker, P.; Bian, J.; Wang, F.: Federated learning for healthcare informatics. J. Healthc. Inf. Res. **5**(1), 1–19 (2021)
178. Mothukuri, V.; Parizi, R.M.; Pouriyeh, S.; Huang, Y.; Dehghantanha, A.; Srivastava, G.: A survey on security and privacy of federated learning. Fut. Gener. Comput. Syst. **115**, 619–640 (2021)
179. Zhang, C.; Xie, Y.; Bai, H.; Yu, B.; Li, W.; Gao, Y.: A survey on federated learning. Knowl. Based Syst. **216**, 106775 (2021)
180. Bin-Hureib, E.; Gutub, A.: Enhancing medical data security via combining elliptic curve cryptography with 1-LSB and 2-LSB image steganography. Int. J. Comput. Sci. Netw. Secur. (IJCSNS) **20**(12), 232–241 (2020). https://doi.org/10.22937/IJCSNS.2020.20.12.26
181. Pandey, S.R.; Tran, N.H.; Bennis, M.; Tun, Y.K.; Manzoor, A.; Hong, C.S.: A crowdsourcing framework for on-device federated learning. IEEE Trans. Wirel. Commun. **19**(5), 3241–3256 (2020)
182. Yu, T.; Bagdasaryan, E.; Shmatikov, V.: Salvaging federated learning by local adaptation (2020). arXiv:2002.04758
183. Bin-Hureib, E.; Gutub, A.: Enhancing medical data security via combining elliptic curve cryptography and image steganography. Int. J. Comput. Sci. Netw. Secur. (IJCSNS) **20**(8), 1–8 (2020). https://doi.org/10.22937/IJCSNS.2020.20.08.1
184. Hardy, S.; Henecka, W.; Ivey-Law, H.; Nock, R.; Patrini, G.; Smith, G.; Thorne, B.: Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. pp. 1–60 (2017)
185. Liu, Y.; Kang, Y.; Xing, C.; Chen, T.; Yang, Q.: A secure federated transfer learning framework. IEEE Intell. Syst. **35**(4), 70–82 (2020)
186. Al-Roithy, B.; Gutub, A.: Remodeling randomness prioritization to boost-up security of RGB image encryption. Multimed. Tools Appl. (MTAP) **80**(18), 28521–28581 (2021). https://doi.org/10.1007/s11042-021-11051-3
187. Wenliang, D.; Han, Y.S.; Chen, S.: Privacy-preserving multivariate statistical analysis: linear regression and classification. In: SIAM International Conference on Data Mining (SDM), pp. 222–233. SIAM (2004)
188. Nikolaenko, V.; Weinsberg, U.; Ioannidis, S.; Joye, M.; Boneh, D.; Taft, N.: Privacy-preserving ridge regression on hundreds of millions of records. In: IEEE Symposium on Security and Privacy, pp. 334–348 (2013)
189. Zhao, L.; Ni, L.; Hu, S.; Chen, Y.; Zhou, P.; Xiao, F.; Wu, L.: InPrivate digging: enabling tree-based distributed data mining with differential privacy. In: IEEE Conference on Computer Communications (INFOCOM), pp. 2087–2095 (2018)
190. Cheng, K.; Fan, T.; Jin, Y.; Liu, Y.; Chen, T.; Dimitrios, P.; Qiang, Y.: SecureBoost: a lossless federated learning framework. IEEE Intell. Syst. (2021). https://doi.org/10.1109/MIS.2021.3082561
191. Zeng, T.; Semiari, O.; Mozaffari, M.; Chen, M.; Saad, W.; Bennis, M.: Federated learning in the sky: joint power allocation and scheduling with UAV swarms. IEEE International Conference on Communications (ICC), pp. 1–6 (2020)
192. Liu, Y.; James, J.Q.; Kang, J.; Niyato, D.; Zhang, S.: Privacy-preserving traffic flow prediction: a federated learning approach. IEEE Internet Things J. **7**(8), 7751–7763 (2020)
193. Pappas, C.; Chatzopoulos, D.; Lalis, S.; Vavalis, M.: IPLS: a framework for decentralized federated learning (2021). arXiv:2101.01901
194. Jakub, K.; McMahan, H.B.; Ramage, D.; Richtárik, P.: Federated optimization: distributed machine learning for on-device intelligence (2016). arXiv:1610.02527
195. Smith, V.; Chiang, C.-K.; Sanjabi, M.; Talwalkar, A.: Federated multi-task learning (2018). arXiv:1705.10467
196. Qadir, S.; Quadri, S.M.K.: Information availability: an insight into the most important attribute of information security. J. Inf. Secur. **7**(3), 185–194 (2016)
197. Samonas, S.; Coss, D.: The CIA strikes back: redefining confidentiality, integrity and availability in security. J. Inf. Syst. Secur. **10**(3), 21–45 (2014)
198. Xianjia, Y.; Queralta, J.P.; Heikkonen, J.; Westerlund, T.: An overview of federated learning at the edge and distributed ledger technologies for robotic and autonomous systems. arXiv–2104 (2021)
199. Ghosh, D.; Vogt, A.: Outliers: an evaluation of methodologies. In: Joint statistical meetings, volume 2012 (2012)
200. Gu, T.; Dolan-Gavitt, B.; Garg, S.: BadNets: identifying vulnerabilities in the machine learning model supply chain (2019). arXiv:1708.06733
201. Taddeo, M.; McCutcheon, T.; Floridi, L.: Trusting artificial intelligence in cybersecurity is a double-edged sword. Nat. Mach. Intell. **1**(12), 557–560 (2019)
202. Yi Ding, A.: MEC and cloud security. Wiley 5G Ref: the essential 5G reference online, pp. 1–16 (2019)
203. Chauhan, M.; Malhotra, R.; Pathak, M.; Singh, U.P.: Different aspects of cloud security. Int. J. Eng. Res. Appl. **2**, 864–869 (2012)
204. Sabahi, F.: Virtualization-level security in cloud computing. In: IEEE International Conference on Communication Software and Networks, pp. 250–254 (2011)
205. Ishiguro, K.; Kono, K.: Hardening hypervisors against vulnerabilities in instruction emulators. In: Proceedings of the 11th European workshop on systems security, pp. 1–6 (2018)

206. Ogasawara, J.; Kono, K.: Nioh: hardening the hypervisor by filtering illegal I/O requests to virtual devices. In: Proceedings of the 33rd annual computer security applications conference, pp. 542–552 (2017)
207. Szefer, J.; Lee, R.B.: Architectural support for hypervisor-secure virtualization. ACM SIGPLAN Notices **47**(4), 437–450 (2012)
208. Siami, M.; Motee, N.: Network abstraction with guaranteed performance bounds. IEEE Trans. Autom. Control **63**(10), 3301–3316 (2018)
209. Gentry, C.; et al.: A fully homomorphic encryption scheme, Vol. 20. Stanford University, Stanford (2009)
210. Ogburn, M.; Turner, C.; Dahal, P.: Homomorphic encryption. Procedia Comput. Sci. **20**, 502–509 (2013)
211. Yi, X.; Paulet, R.; Bertino, E.; Homomorphic encryption. In: Homomorphic encryption and applications, pp. 27–46. Springer (2014)
212. Alotaibi, M.; Al-hendi, D.; Alroithy, B.; AlGhamdi, M.; Gutub, A.: Secure mobile computing authentication utilizing hash, cryptography and steganography combination. J. Inf. Secur. Cybercrim. Res. (JISCR) **2**(1), 9–20 (2019). https://doi.org/10.26735/16587790.2019.001
213. Singh, A.; Chatterjee, K.; Satapathy, S. C.: An edge based hybrid intrusion detection framework for mobile edge computing. Complex Intell. Syst., pp. 1–28, 2021.
214. Ierace, N.; Urrutia, C.; Bassett, R.: Intrusion prevention systems. Ubiquity **6**(19), 2–2 (2005)
215. Rengaraju, P.; Raja Ramanan, V.; Lung, C.-H.: Detection and prevention of DoS attacks in software-defined cloud networks. In: IEEE Conference on Dependable and Secure Computing, pp. 217–223 (2017)
216. Wang, L.; Schwing, A.G.; Lazebnik, S.: Diverse and accurate image description using a variational auto-encoder with an additive Gaussian encoding space (2017). arXiv:1711.07068
217. Bengio, Y.: Gradient based optimization of hyper-parameters. Neural Comput. **12**(8), 1889–1900 (2000)
218. Goyal, V.; Tripathy, R.: An efficient solution to the ARP cache poisoning problem. In: Australasian Conference on Information Security and Privacy, pp. 40–51. Springer (2005)
219. Oliveira, R.M.S.; Zaiane, O.R.: Protecting sensitive knowledge by data sanitization. In: IEEE International Conference on Data Mining, pp. 613–616 (2003)
220. Sarasamma, S.T.; Zhu, Q.A.; Huff, J.: Hierarchical Kohonenen net for anomaly detection in network security. IEEE Trans. Syst. Man Cybern. Part B (Cybern.) **35**(2), 302–312 (2005)
221. Trimble, M.: Geoblocking, technical standards and the law (2016)
222. Zhang, X.; Li, C.; Zheng, W.: Intrusion prevention system design. In: IEEE International Conference on Computer and Information Technology, pp. 386–390 (2004)
223. Albright, J.G.: The basics of an IT security policy. GSEC practical requirement V. 1.3 SANS Institute of Technology, 1 (2002)
224. Abdulmohsin, I.: Techniques and algorithms for access control list optimization. Comput. Electr. Eng. **35**(4), 556–566 (2009)
225. Fung, C.J.; McCormick, B.: Vguard: a distributed denial of service attack mitigation method using network function virtualization. In: IEEE International Conference on Network and Service Management (CNSM), pp. 64–70 (2015)
226. Ryoo, J.; Rizvi, S.; Aiken, W.; Kissell, J.: Cloud security auditing: challenges and emerging approaches. IEEE Secur. Priv. **12**(6), 68–74 (2013)
227. Takebayashi, T.; Tsuda, H.; Hasebe, T.; Masuoka, R.: Data loss prevention technologies. Fujitsu Sci. Tech. J. **46**(1), 47–55 (2010)
228. Kaufman, L.M.: Data security in the world of cloud computing. IEEE Secur. Privacy **7**(4), 61–64 (2009)
229. Reddy, T.A.; Saman, N.F.; Claridge, D.E.; Haberl, J.S.; Dan Turner, W.; Chalifoux, A.T.: Baselining methodology for facility-level monthly energy use-part 1: theoretical aspects. In: ASHRAE Transactions, pp. 336–347. ASHRAE (1997)
230. Lindner, M.; McDonald, F.; McLarnon, B.; Robinson, P.: Towards automated business-driven indication and mitigation of VM sprawl in Cloud supply chains. In: 12th IFIP/IEEE International Symposium on Integrated Network Management (IM 2011) and Workshops, pp. 1062–1065 (2011)
231. Atzeni, I.; Luis, G.; Scutari, G.; Palomar, D.P.; Fonollosa, J.R.: Demand-side management via distributed energy generation and storage optimization. IEEE Trans. Smart Grid **4**(2), 866–876 (2012)
232. Chandramouli, R.: Security recommendations for hypervisor deployment on servers. NIST Spec. Publ. **800**, 125A (2018)
233. Deri, L.; Martinelli, M.; Cardigliano, A.: Realtime high-speed network traffic monitoring using ntopng. In: 28th large installation system administration conference (LISA14), pp. 78–88 (2014)
234. Jansen, W.A.: Cloud hooks: security and privacy issues in cloud computing. In: IEEE Hawaii International Conference on System Sciences, pp. 1–10 (2011)
235. Pawar, D.; Geethakumari, G.: Digital forensic architecture for cloud computing systems: methods of evidence identification, segregation, collection and partial analysis. In: Information Systems Design and Intelligent Applications, pp. 213–225. Springer (2016). https://doi.org/10.1007/978-81-322-2755-7_22
236. Dasgupta, D.; Roy, A.; Nag, A.: Multi-factor authentication. pp. 185–233 (2017)
237. Jablon, D.P.: Strong password-only authenticated key exchange. ACM SIGCOMM Comput. Commun. Rev. **26**(5), 5–26 (1996)
238. Alassaf, N.; Gutub, A.: Simulating light-weight-cryptography implementation for IoT healthcare data security applications. Int. J. E-Health Med. Commun. (IJEHMC) **10**(4), 1–15 (2019). https://doi.org/10.4018/IJEHMC.2019100101
239. Alassaf, N.; Gutub, A.; Parah, S.A.; Al Ghamdi, M.: Enhancing speed of SIMON: a light-weight-cryptographic algorithm for IoT applications. Multimed. Tools Appl. **78**(23), 32633–32657 (2019). https://doi.org/10.1007/s11042-018-6801-z
240. Yubin, G.; Liankuan, Z.; Fengren, L.; Ximing, Li.: A solution for privacy-preserving data manipulation and query on NoSQL database. J. Comput. **8**(6), 1427–1432 (2013)
241. Deswarte, Y.; Quisquater, J.-J.; Sa¨ıdane, A.: Remote integrity checking. In: Working Conference on Integrity and Internal Control in Information Systems, pp. 1–11. Springer (2003)
242. Peddoju, S.K.; Upadhyay, H.; Lagos, L.: File integrity monitoring tools: Issues, challenges, and solutions. Concurr. Comput. Pract. Exp. **32**(22), e5825 (2020)
243. Kent, K.; Souppaya, M.: Guide to computer security log management. NIST Spec. Publ. **92**, 1–72 (2006)
244. Scholte, T.; Robertson, W.; Balzarotti, D.; Kirda, E.: Preventing input validation vulnerabilities in web applications through automated type analysis. IEEE Annual Computer Software and Applications Conference, pp. 233–243 (2012)
245. Buehrer, G.; Weide, B.W.; Sivilotti, P.A.G.: Using parse tree validation to prevent SQL injection attacks. In: Proceedings of the 5th International Workshop on Software Engineering and Middleware, pp. 106–113 (2005)
246. Ntagwabira, L.; Kang, S.L.: Use of query tokenization to detect and prevent SQL injection attacks. IEEE Int. Conf. Comput. Sci. Inf. Technol. **2**, 438–440 (2010)
247. Pietraszek, T.; Berghe, C.V.: Defending against injection attacks through context-sensitive string evaluation. In: International Workshop on Recent Advances in Intrusion Detection, pp. 124–145. Springer (2005)
248. Gossweiler, R.; Kamvar, M.; Baluja, S.; What's up CAPTCHA? A CAPTCHA based on image orientation. In: Proceedings of the 18th International Conference on World Wide Web, pp. 841–850 (2009)

249. Singh, A.; Chatterjee, K.: A secure multi-tier authentication scheme in cloud computing environment. In: IEEE Conference on Circuits, Power and Computing Technologies (ICCPCT), pp. 1–7 (2015)
250. Ioannidis, S.; Keromytis, A.D.; Bellovin, S.M.; Smith, J.M.: Implementing a distributed firewall. In: ACM Conference on Computer and Communications Security, pp. 190–199 (2000)
251. Venema, W.: TCP wrapper: network monitoring, access control, and booby traps. In: UNIX Security Symposium III: proceedings: Baltimore, MD, September 14–16, p. 85 (1992)
252. Sokol, P.; Misek, J.; Husak, M.: Honeypots and honeynets: issues of privacy. EURASIP J. Inf. Secur. **2017**(1), 1–9 (2017)
253. Shambour, M.; Gutub, A.: Personal privacy evaluation of smart devices applications serving Hajj and Umrah rituals. J. Eng. Res. **1**, 2 (2021). https://doi.org/10.36909/jer.13199
254. Long, D.D.E.; Montague, B.R.; Cabrera, L.-F.: Swift/RAID: a distributed RAID system. Comput Syst **7**(3), 333–359 (1994)
255. Tahboub, R.; Saleh, Y.: Data leakage/loss prevention systems (DLP). In: IEEE World Congress on Computer Applications and Information Systems (WCCAIS), pp. 1–6 (2014)
256. Kheshaifaty, N.; Gutub, A.: Preventing multiple accessing attacks via efficient integration of captcha crypto hash functions. Int. J. Comput. Sci. Netw. Secur. (IJCSNS) **20**(9), 16–28 (2020). https://doi.org/10.22937/IJCSNS.2020.20.09.3
257. Singh, A.; Chandra, U.; Kumar, S.; Chatterjee, K.: A secure access control model for e-health cloud. In: IEEE Region 10 Conference (TENCON), pp. 2329–2334 (2019)
258. Bijalwan, A.; Wazid, M.; Pilli, E.S.; Joshi, R.C.: Forensics of random-UDP flooding attacks. J. Netw. **10**(5), 287 (2015)
259. Verma, K.; Hasbullah, H.; Kumar, K.: An efficient defense method against UDP spoofed flooding traffic of denial of service (DoS) attacks in VANET. In: IEEE International Advance Computing Conference (IACC), pp. 550–555 (2013)
260. Gupta, N.; Jain, A.; Saini, P.; Gupta, V.: DDoS attack algorithm using ICMP flood. In: IEEE International Conference on Computing for Sustainable Global Development (INDIACom), pp. 4082–4084 (2016)
261. Saad, R.M.A.; Almomani, A.; Altaher, A.; Gupta, B.B.; Manickam, S.: ICMPv6 flood attack detection using DENFIS algorithms. Indian J. Sci. Technol **7**(2), 168 (2014)
262. Bogdanoski, M.; Suminoski, T.; Risteski, A.: Analysis of the SYN flood DoS attack. Int. J. Comput. Netw. Inf. Secur. (IJCNIS) **5**(8), 1–11 (2013)
263. Haris, S.H.C.; Ahmad, R.B.; Ghani, M.A.H.A.: Detecting TCP SYN flood attack based on anomaly detection. In: IEEE International Conference on Network Applications, Protocols and Services, pp. 240–244 (2010)
264. Harris, B.; Hunt, R.: TCP/IP security threats and attack methods. Comput. Commun. **22**(10), 885–897 (1999)
265. Bilge, L.; Dumitras, T.: Before we knew it: an empirical study of zero-day attacks in the real world. In: Proceedings of the 2012 ACM conference on Computer and communications security, pp. 833–844 (2012)
266. Kumar, A.: Zero day exploit. Available at SSRN 2378317 (2014)
267. Biggio, B.; Nelson, B.; Laskov, P.: Poisoning attacks against support vector machines (2013). arXiv:1206.6389
268. Zhang, X.; Zhu, X.; Lessard, L.: Online data poisoning attack. PMLR learning for dynamics and control, pp. 201–210 (2020)
269. Biggio, B.; Corona, I.; Maiorca, D.; Nelson, B.; Srndi´c, N.; Laskov, P.; Giacinto, G.; Roli, F.: Evasion attacks against machine learning at test time. Joint European conference on machine learning and knowledge discovery in databases, pp. 387–402. Springer (2013)
270. Zhang, F.; Chan, P.P.K.; Biggio, B.; Yeung, D.S.; Roli, F.: Adversarial feature selection against evasion attacks. IEEE Trans. Cybernet. **46**(3), 766–777 (2015)
271. Gutub, A.; Al-Roithy, B.: Varying PRNG to improve image cryptography implementation. J. Eng. Res. **9**(3A), 153–183 (2021). https://doi.org/10.36909/jer.v9i3A.10111
272. Hassan, F.; Gutub, A.: Improving data hiding within colour images using hue component of HSV colour space. CAAI Trans. Intell. Technol. IET (IEE) (2021). https://doi.org/10.1049/cit2.12053
273. Aono, T.; Higuchi, K.; Ohira, T.; Komiyama, B.; Sasaoka, H.: Wireless secret key generation exploiting reactance-domain scalar response of multipath fading channels. IEEE Trans. Antennas Propag. **53**(11), 3776–3784 (2005)
274. Ball, J.; Dragan, A.; Banaszek, K.: Exploiting entanglement in communication channels with correlated noise. Phys. Rev. A **69**(4), 042324 (2004)
275. Halfond, W.G.; Viegas, J.; Orso, A.; et al.: A classification of SQL injection attacks and countermeasures s. In: IEEE international Symposium on Secure Software Engineering, vol. 1, pp. 13–15 (2006)
276. Kieyzun, A.; Guo, P.J.; Jayaraman, K.; Ernst, M.D.: Automatic creation of SQL injection and cross-site scripting attacks. In: IEEE International Conference on Software Engineering, pp. 199–209 (2009)
277. Shar, L.K.; Tan, H.B.K.; Briand, L.C.: Mining SQL injection and cross site scripting vulnerabilities using hybrid program analysis. In: IEEE International Conference on Software Engineering (ICSE), pp. 642–651 (2013)
278. De Ryck, P.; Desmet, L.; Joosen, W.; Piessens, F.: Automatic and precise client-side protection against CSRF attacks. In: European Symposium on Research in Computer Security, pp. 100–116. Springer (2011)
279. Barth, A.; Jackson, C.; Mitchell, J.C.: Robust defenses for cross-site request forgery. In: ACM Conference on Computer and Communications Security, pp. 75–88 (2008)
280. Jablon, D.P.: Extended password key exchange protocols immune to dictionary attack. In: IEEE Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises, pp. 248–255 (1997)
281. Vykopal, J.; Plesnik, T.; Minarik, P.: Network-based dictionary attack detection. In: IEEE International Conference on Future Networks, pp. 23–27 (2009)
282. Hassan Adnan, A.; Abdirazak, M.; Shamsuzzaman Sadi, A.B.M.; Anam, T.; Zaman Khan, S.; Rahman, M.M.; Omar, M.M.: A comparative study of WLAN security protocols: WPA, WPA2. In: IEEE international conference on advances in electrical engineering (ICAEE), pp. 165–169 (2015)
283. Rumale, A.S.; Chaudhari, D.: IEEE 802. 11 x , and WEP , EAP , WPA / WPA 2. Tech. Appl, 2(6):1945–1950, 2011
284. Hammer-Lahav, E.; Recordon, D.; Hardt, D.: The OAuth 1.0 Protocol. Technical report, RFC 5849, April, 2010
285. Hardt, D.; et al.: The OAuth 2.0 authorization framework (2012)
286. Johns, M.; Braun, B.; Schrank, M.; Posegga, J.: Reliable protection against session fixation attacks. ACM Symposium on Applied Computing, pp. 1531–1537 (2011)
287. Kolšek, M.: Session fixation vulnerability in web-based applications. Acros Secur. **1**, 1–15 (2002)
288. Chen, E.Y.; Pei, Y.; Chen, S.; Tian, Y.; Kotcher, R.; Tague, P.; OAuth demystified for mobile application developers. In: CCS'14: ACM SIGSAC Conference on Computer and Communications Security, pp. 892–903 (2014)
289. Xu, X.; Wang, L.; Youssef, A.; Zhu, B.: Preventing collusion attacks on the one-way function tree (OFT) scheme. In: International Conference on Applied Cryptography and Network Security, pp. 177–193. Springer (2007)
290. Joyia, G.J.; Liaqat, R.M.; Farooq, A.; Rehman, S.: Internet of medical things (IOMT): applications, benefits and future challenges in healthcare domain. J. Commun. **12**(4), 240–247 (2017)

291. Magsi, H.; Sodhro, A.H.; Chachar, F.A.; Abro, S.A.K.; Sodhro, G.H.; Sandeep, P.: Evolution of 5G in internet of medical things. In: IEEE International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), pp. 1–7 (2018)

292. Samkari, H.; Gutub, A.: Protecting medical records against cybercrimes within hajj period by 3-layer security. Recent Trends Inf. Technol. Appl. **2**(3), 1–21 (2019). https://doi.org/10.5281/zenodo.3543455