Cybersecurity, Cloud and Edge Computing*

1st Shashwat Prakash
Computer Science Engineering
Shiv Nadar Institution of Eminence
2010111108

2rd Shashwat Tiwari
Computer Science Engineering
Shiv Nadar Institution of Eminence
2010111038

3nd Shrey Sharma
Computer Science Engineering
Shiv Nadar Institution of Eminence
2010110600

***Abstract*—This paper gives us a brief overlook into the risk evaluation of cloud computing, Block design-based key agreement for data sharing in cloud computing, Threats, risks, opportunities of quantum cybersecurity, efficient computing resources sharing for mobile edge-cloud computing networks, and edge computing's visions and challenges.**

***Index Terms*—Cloud computing, edge computing, quantum cybersecurity.**

I. Threat-Specific Security Risk Evaluation in the Cloud

Cloud computing refers to delivering computing services, including servers, storage, databases, software, and more, over the internet. Rather than relying on local servers or personal devices for data storage and processing, cloud computing allows users to access these resources from anywhere with an internet connection. This provides a more flexible, scalable, and cost-effective solution for businesses and individuals who need to manage large amounts of data and applications.

The nature of attacks on cloud computing is diverse and depends on what attackers hope to achieve, what type of resources they want to access, and what threats they pose. In multi-tenant cloud computing environments, a generic security model may not be adequate to meet the varied security needs and requirements of different clients. For a specific client, service availability may be a security requirement expected to be ensured by the cloud provider, whereas for another client, data integrity might be more important than other security objectives.

Using a 'one-size-fits-all' approach for evaluating security risks in cloud-based computing for individual clients is inappropriate. This can lead to inaccurate risk assessment and less effective mitigation strategies. The limitations of these approaches for risk evaluation are two-fold: (1) they do not consider the specific security requirements of individual clients, resulting in inaccurate risk assessment, and (2) as a consequence, security administrators may implement less effective mitigation strategies based on the flawed assessment. Clearly, some important aspects of current risk evaluation approaches are missing. Each client of a cloud service needs to be guaranteed that its major security requirements are met. The key question to be answered is, "how does the cloud provider assess the security risk associated with individual enterprises based on their specific security requirements?"

Identify applicable funding agency here. If none, delete this.

Our proposed risk evaluation approach is tailored to address specific threats and includes the following characteristics:

- It centers on the unique security needs of individual clients in relation to their use of cloud-based computing.
- It prioritizes implementing security solutions that meet the specific security requirements of each client.
- It calculates and devises effective security solutions for deployment in the cloud based on the results of threat-specific risk evaluation.

The proposed technique employs vulnerability information to evaluate security risks from the perspective of specific threats to a client's security requirements. This empowers cloud providers' security administrators to devise effective mitigation strategies that address perceived threats to a client's computing assets. The technique also facilitates prioritizing threats based on severity, as computed by the proposed risk assessment technique. The computation process considers the relative importance of identified threats, the likelihood of an attack, and its potential impact.

One notable feature of our threat-specific risk evaluation approach is that it allows security administrators to identify threats relevant to a particular client and assess only the security risks associated with those threats. This leads to a more precise security risk evaluation and a more appropriate allocation of mitigation resources. The technique is not restricted to cloud computing alone but applies to other networked systems.

The main contributions of the paper [1] are as follows:

- A threat-specific risk evaluation based on vulnerability information, which assesses the security risk of specific threats to an asset by analyzing security attributes of vulnerabilities in the network.
- A threat-guided countermeasure selection selects the optimal countermeasure from a pool of different application options based on the specific threat(s) identified.
- A threat-specific security risk evaluation software tool is a prototype tool designed to evaluate the security risk of the cloud, considering different threat categories.
- An experimental analysis that demonstrates the applicability, feasibility, and importance of the proposed approach through simulations.

II. Block Design-Based Key Agreement for Group Data Sharing in Cloud Computing

A. Introduction

We prefer to store all types of data in cloud servers right now because of the limited storage resources available and the need for easy access. This is a good option for businesses and organizations because it saves them the expense of setting up and maintaining equipment when data is stored locally. The cloud server offers people and businesses an accessible and practical storage platform, but it also poses security risks.

To maintain the security of their subsequent communications, many participants create a single conference key using a key agreement mechanism, which can be deployed to promote efficient and secure data sharing in the cloud.

The Diffie-Hellman protocol efficiently solves the problem of creating a shared secret key between two parties in cryptography. This type of key agreement protocol allows both parties to contribute to creating the key, which can then be used to secure their communication. A secure key agreement protocol ensures that attackers cannot obtain the key through malicious means like eavesdropping. This makes it useful in high-security communication environments where interactive communication is required.

The Diffie-Hellman key agreement protocol generates keys but lacks an authentication service, leaving it vulnerable to man-in-the-middle attacks. To address this limitation, authentication mechanisms can be added to the protocol. Another limitation of the Diffie-Hellman protocol is that it can only support two participants.

B. Main Contribution of the paper

By extending the SBIBD's structure to support more participants, a block design-based key agreement system that is both effective and secure can be created.

This makes it possible for different data owners to securely and effectively exchange the outsourced data. To allow group data sharing in cloud computing, the SBIBD is built as the group data sharing model. Moreover, the protocol can offer fault tolerance and authentication services.

The SBIBD [2] structure is used to develop a model of group data exchange. Based on the SBIBD definition, a group data-sharing model is created in this study and can be used to decide how members will communicate with one another. General methods for calculating the common conference key for numerous participants are obtained from mathematical representations of the SBIBD structure.

The protocol may include fault tolerance and fault detection. To ensure that a shared conference key is successfully created among all participants, the proposed protocol can perform fault detection. To support the fault tolerance property, a volunteer will be utilized to substitute a malicious participant during the fault detection phase. The volunteer makes the protocol resistant to certain key attacks, which increases the security of group data sharing in cloud computing.

The protocol can facilitate safe group data exchange in cloud computing. The SBIBD-based data-sharing concept suggests that several players can collaborate to share the outsourced data effectively. The security of the outsourced group data is ensured by each group member performing the key agreement to generate a shared conference key.

Take note that only group members can generate the shared conference key. The created key is not accessible to attackers or the semi-trusted cloud server. As a result, they cannot access the actual data that was outsourced (and instead only receive some gibberish). The suggested key agreement protocol allows efficient and safe group data sharing in cloud computing.

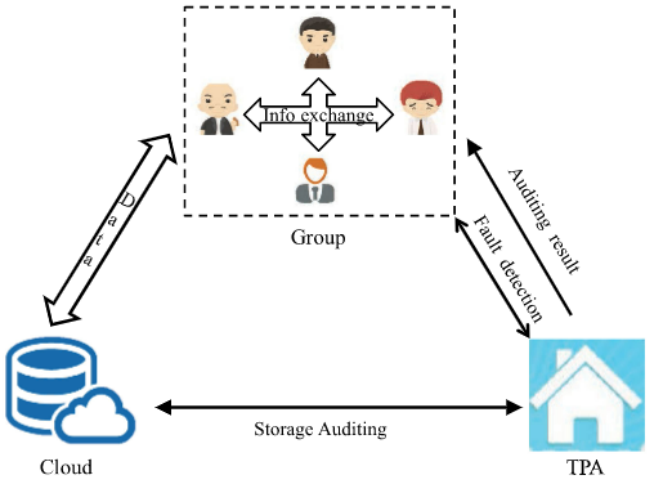

Fig. 1. System model of data sharing in cloud computing.

The contributions mentioned earlier have significantly expanded the potential uses of the key agreement protocol by implementing a highly secure and adaptable SBIBD. Additionally, our protocol reduces communication complexity without adding any computational complexity. In particular, our protocol has a communication complexity of $O(n\sqrt{n})$ and a computational complexity of $O(nm^2)$. Here, n refers to the number of participants, and m is the extension degree of Fpm, which represents the rational points in a supersingular elliptic curve.

III. Review of Quantum Cybersecurity: Threats, Risks and Opportunities

A. Introduction

The increasing digitization of sensitive data in databases highlights the importance of cybersecurity in protecting against cyber threats. As cyber-attacks become more frequent, the field of cybersecurity is looking to embrace futuristic technologies like AI, Quantum Computing, Blockchain, and Data Science. However, the potential risks and opportunities of Quantum Computing on cybersecurity must be investigated to prevent current cybersecurity infrastructure from becoming obsolete. This paper aims to study the intersection between Quantum Computing and Cybersecurity and explore ways to enhance Quantum Computing cybersecurity for concerned stakeholders.

B. Paper Organization

This section outlines the organization of the paper and its various sections. I. Introduction This section introduces the paper and its focus on quantum cybersecurity. II. Research Design and Methodology This section explains the research design and methodology used for the paper, including research questions on both positive and negative aspects of quantum cybersecurity. III. Overview of Quantum Computing and Cybersecurity This section provides a thorough overview of quantum computing and cybersecurity. IV. Related Studies This section reviews 20 primary studies on quantum computing and cybersecurity and synthesizes the findings, approaches, and considerations. V. Findings and Future Research Directions This section discusses the findings and challenges identified from the review and provides guidelines for future research in this field. VI. Conclusion This section concludes the paper and summarizes its main contributions.

C. Research Design and Methodology

1) Research Goals: This study aims to address three key research questions related to the intersection of quantum computing and cybersecurity. The first research question seeks to define quantum computing and how it can intersect with cybersecurity. The second research question evaluates quantum cybersecurity's potential opportunities and risks. Finally, the third research question explores the improvements that can be carried out in quantum cybersecurity. The authors recognise the threat posed by the development of quantum computing to cybersecurity and aim to provide insights that can improve the understanding of this intersection.

2) Study Selection: To identify research papers relevant to the study, the authors implemented a search process using various scientific databases. They prepared potential search strings containing keywords related to the study topic, including "Quantum Cybersecurity," "Quantum Computing for Cybersecurity," "Cybersecurity for Quantum Computing," "Quantum-enabled Cybersecurity," and "Post-Quantum Cybersecurity." The authors then applied specific search strings using the selected keywords and the title of the papers during the analysis. The search was conducted on March 01st, 2022, and included all studies published up to that date. The scientific databases used for procuring these papers included Google Scholar, IEEE Xplore Digital Library, ScienceDirect, ACM Digital Library, SpringerLink, and arXiv e-Print Archive. The authors used a systematic literature review process and depicted the attrition of the literature through processing in Fig.

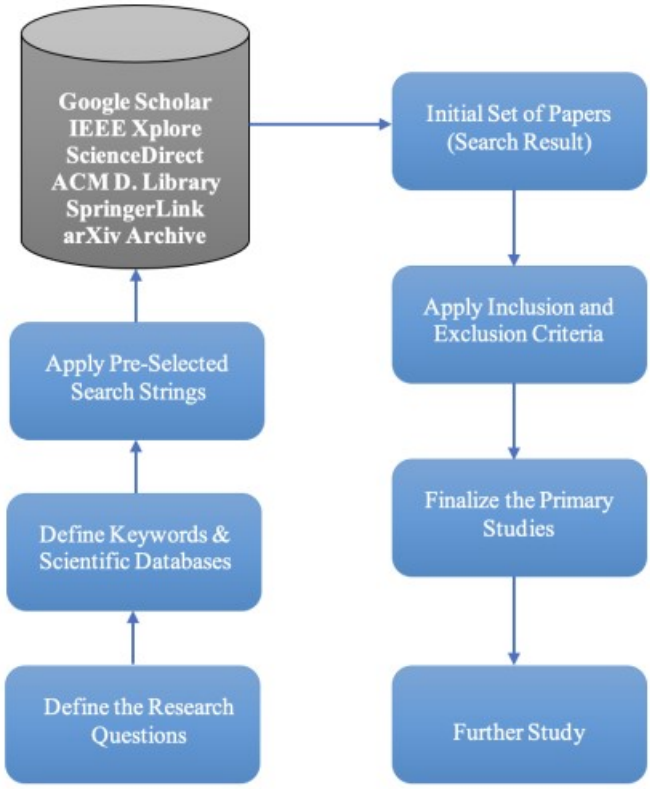

Fig. 1. Attrition of Systematic Literature thorough processing

3) Results of the Studies: After completing the initial search, a total of 354 research papers were identified. However, the authors applied the inclusion and exclusion criteria to ensure that only relevant papers were included in the study. This process excluded 329 papers, leaving 25 papers for further review. The authors then conducted a full-text review of these 25 papers and excluded four more papers that did not meet the inclusion criteria. Thus, a total of 21 papers were finally included in the study. The authors used a PRISMA flow diagram to depict the selection process presented in Fig.

D. Related Studies

The text discusses a few research papers on post-quantum cryptography. Some of the papers that have been mentioned are as follows: One paper proposes five types of post-quantum cryptosystems for blockchain technology. The paper suggests that rapid advancement in quantum computing using Grover's and Shor's algorithms puts hash functions and public-key cryptography at risk. The proposed cryptosystems are resilient to quantum threats and can be used for encryption and digital signatures in blockchain. Another paper suggests lattice-driven cryptographic techniques for quantum resistance security mod-

els in the Internet of Things (IoT). The paper analyzes the vulnerabilities of current IoT architecture and implementation and proposes improvements to protect IoT in the post-quantum era. A third paper proposes KeyShield, a scalable and quantum-safe key management scheme that outperforms state-of-the-art schemes in quantum-resistance, computation cost, message overhead, storage cost, and rekeying delay. The paper provides mathematical analysis and proofs to support the proposed scheme.

E. Conclusion

The emergence of quantum computing can significantly impact cybersecurity as it can be used both as a threat and a solution. The study reviewed the current state-of-the-art in quantum computing and cybersecurity and identified proposed approaches. Quantum computing can be used to enhance cybersecurity, but it also poses threats to cybersecurity. The review highlights the need for further research in this emerging field and provides insights into future research directions for quantum and cybersecurity practitioners and researchers.

IV. Efficient Computing Resource Sharing for Mobile Edge-Cloud Computing Networks

A. Introduction

The demand for mobile applications has brought challenges to designing mobile devices with limited hardware capabilities. The cloud and edge can provide computing services to address this challenge, but designing an efficient cooperative edge-cloud network and business model is necessary for the future. Our previous work proposed an efficient wholesale and buyback scheme for managing computing resources in mobile edge-cloud computing networks to maximise profitability and QoS. However, the scheme did not consider the impact of cloud operation costs and computing requirements on wholesale prices. Therefore, we propose a joint pricing scheme for cloud computing resources to determine optimal computing resources for both the MEC and cloud. The scheme emphasizes the importance of collaboration between the cloud and edge servers and highlights the potential benefits for both entities. The rest of the paper introduces related works, proposes a framework for mobile edge-cloud computing networks, and formulates computing resource management problems. The paper concludes with simulations demonstrating the efficiency of the proposed algorithm.

B. Related Works

The related works discussed in this paper can be categorised into two perspectives: system design and system optimisation to enhance MEC systems. For instance, a MEC-based object detection architecture was proposed for real-time surveillance applications, while an air-ground integrated mobile edge network (AGMEN) using UAVs was proposed to improve the edge's communication, caching, and computing network. On the other hand, from a system optimisation perspective, researchers have proposed schemes to optimise different aspects of MEC systems, such as energy consumption, workload allocation, and resource allocation. For example, an energy-efficient computing offloading management scheme was proposed to reduce the energy consumption of mobile devices. In contrast, an optimal workload allocation scheme was proposed to balance power consumption and transmission delay. Moreover, a joint radio and computational resource allocation scheme were designed to enhance system performance and improve user satisfaction. In addition to these approaches, some studies proposed heuristic solutions to address system design and optimisation problems.

C. System model and problem formulation

The problem formulation in this system model can be described as a resource allocation problem. The goal is to optimize the utilization of computing resources between the MEC servers and the cloud servers. The objective is to maximize the overall profit of the system while ensuring the QoS requirements of the computing tasks. To achieve this objective, the system must determine the optimal allocation of computing resources between the MEC servers and the cloud servers, considering the current computing requirements, available computing resources, and the costs of computing resource wholesale and buyback. The problem is further complicated by the time-sensitive nature of the computing tasks and the need to ensure QoS. The resource allocation problem can be mathematically formulated as an optimization problem, where the objective function is to maximize the overall profit of the system, subject to constraints such as the QoS requirements, resource availability, and computing resource costs. The optimization problem can be solved using various optimization techniques, such as linear programming, dynamic programming, or heuristic algorithms.

1) Operation Model of Mobile Edge Computing (MEC): The MEC server's computing resources are divided into two parts, one reserved for local computation tasks and the other wholesaled to the cloud for flexible computation tasks. Communication delay between MEC servers and the cloud is negligible due to wired connections and local processing.

2) Operation Model of Cloud Computing: Cloud computing resources are available from cloud servers and MEC servers, managed by the cloud. The wholesale and buyback scheme of MEC servers affects the availability of computing resources from MEC servers.

D. Profit Model of MEC and Cloud Computing

The profit of MEC servers includes operation cost, income from local processing, income from wholesaling computing resources to the cloud, and cost for buying back computing resources. The total profit depends on the wholesaled and buyback scheme of the MEC server. The profit of the cloud includes processing computing tasks, local operation cost, and trading computing resources with MEC servers. The cloud needs to balance operation costs and QoS penalty.

E. Wholesale and Buyback with Profit

The MEC and cloud have different profit objectives but can share computing resources with profit transfers. The MEC

aims to maximize profit by providing computing services and wholesaling computing resources, while the cloud aims to provide better computing services and reduce operation costs. MEC servers determine their wholesaled and buyback scheme, while the cloud determines the wholesale price and manages cloud computing resources. These problems are coupled by profit transfers, and optimal pricing and resource management are required to maximize the profits of both MEC and cloud.

F. Conclusion

This paper proposed a framework for efficient resource sharing between the MEC and the cloud to improve their profitability. Two cases were considered: social welfare maximization and respective profit maximization. For the first case, it was proved that the social welfare only depends on the cloud computing resources and the concavity of the social welfare maximization problem. For the second case, an optimal pricing and cloud computing resource management were designed to maximize the total profit. Numerical evaluations showed that the proposed algorithms can maximize the social welfare and the respective profits of the MEC and the cloud separately. However, the assumptions made in this paper are that all MEC servers have the same computing resources and all computing tasks have similar QoS requirements.

V. Edge Computing: Vision and Challenges

A. Introduction

The Internet of Things (IoT) was first introduced to the community in 1999 for supply chain management. Then the concept of "making a computer sense information without the aid of human intervention" was widely adapted to other fields such as healthcare, home, environment, and transport. Now with IoT, we will arrive in the post-cloud era, where there will be a large quality of data generated by things immersed in our daily lives. A lot of applications will also be deployed at the edge to consume this data. ... Several case studies like cloud offloading, smart home, and city, as well as collaborative edge, are introduced to explain edge computing in a detailed manner further, followed by some challenges and opportunities in programmability, naming, data abstraction, service management, privacy, and security, as well as optimization metrics that are worth future research and study.

B. Significance of Edge Computing

Edge computing is the enabling technology that allows computation at the network's edge between data sources and cloud data centers. It is interchangeable with fog computing, but edge computing focuses more on the side of the thing, while fog computing focuses more on the infrastructure side. Edge can perform computing offloading, data storage, caching, and processing, and distribute requests and delivery service from cloud to user. It needs to be well-designed to meet the reliability, security, and privacy protection requirement.

- Push from IoT: Data must be processed at the edge for shorter response times, more efficient processing, and smaller network pressure. Current network bandwidth and reliability are challenged for supporting many vehicles in one area.
- Pull from IoT: IoT is a growing network of electrical devices that will consume raw data at the edge, leading to unnecessary bandwidth and computing resource usage. Offloading some computing tasks to the edge could be more energy efficient.
- Change From Data Consumer to Producer: The change from a data consumer to a data producer/consumer requires more function placement at the edge, such as demising and adjusting video clips before uploading to the cloud. This is especially important for wearable health devices, as processing the data at the edge can protect user privacy better than uploading raw data.*

C. Case Study

- Cloud Offloading: Edge computing provides a way to offload part of the workload from the cloud, allowing for faster data processing and improved user experience. Edge computing can reduce latency and improve user experience for time-sensitive applications, such as navigation, content filtering/aggregating, and real-time applications.

- Video Analytics: Video analytics is becoming increasingly important due to the long data transmission latency and privacy concerns. Edge computing can leverage the data and computing power on every thing to get faster results.
- Smart Home: Edge computing is ideal for building a smart home, as it can be connected and managed easily, processed locally, and deployed on the edgeOS for better management and delivery.
- Smart City: Edge computing is an ideal platform for smart cities due to its close proximity to data sources.
- Collaborative Edge: Collaborative Edge is a physical small data center that connects cloud and end-user with data processing capabilities, allowing stakeholders to share and cooperate data. Collaborative edge enables pharmacies to provide purchasing records to hospitals, retrieve the population of the flu outbreak, and reschedule production plans and rebalance inventories.

D. Challenges and Opportunities

- Programmability: The concept of computing stream is proposed to address the programmability of edge computing, allowing data to be processed in distributed and efficient fashion.
- Naming: Edge computing needs an efficient naming scheme to handle the mobility of things, highly dynamic network topology, privacy and security protection, and unreliable things. IP based naming schemes are not flexible enough to serve the dynamic edge network, so new naming mechanisms such as named data network- ing (NDN) and MobilityFirst can be used. EdgeOS assigns a unique human friendly name to each thing, making management easy and providing better programmability to service providers.
- Data Abstraction: Edge computing should minimize human involvement and consume/process all data at the gateway level, with processed data sent to the upper layer for future service providing. Data abstraction is necessary for privacy and security, but it can be difficult to decide the degree of abstraction.
- Service management at the edge of the network should have four fundamental features: differentiation, extensibility, isolation, and reliability. A well-designed control access mechanism should be added to the service management layer to improve reliability at the edge of the network.
- Privacy and Security: At the edge of the network, usage privacy and data security are important services, but how to support service without harming privacy is a challenge. The most important idea is that end user data collected at the edge of the network should be stored at the edge and the user should be able to control if the data should be used by service providers.

E. Optimization Metrics

Optimization metrics such as latency, bandwidth, energy and cost are important to evaluate performance in edge computing. High bandwidth can reduce transmission time, improve latency, and save bandwidth between the edge and the cloud.Edge computing requires a tradeoff between computation energy consumption and transmission energy consumption. Edge computing provides less latency and energy consumption, increased throughput, and improved user experience.

F. Conclusion

Edge computing is becoming increasingly important due to the IoT and universalized mobile devices, making it more efficient to process or massage data at the edge of the network. This paper outlines several cases of edge computing, such as cloud offloading to a smart environment, collaborative edge, programmability, naming, data abstraction, service management, privacy and security, and optimization metrics. It is hoped this paper will bring this to the attention of the community.

VI. Mobile Edge Computing: A Survey on Architecture and Computation Offloading

A. Introduction

Mobile cloud computing (MCC) provides advantages such as extending battery lifetime, enabling sophisticated applications, and providing higher data storage capabilities, but also introduces high latency. Edge computing provides significantly lower latencies and jitter than the MCC, while the cloudlet concept provides limited computational and storage resources. Fog computing is a key enabler of Internet of Things (IoT) and big data applications due to its low latency, widespread geographical distribution, interconnection of nodes, and support of streaming and real time applications. Cloud Radio Access Network (C-RAN) and Mobile Edge Computing (MEC) are two concepts integrating the cloud capabilities into the mobile network. This paper discusses the heterogeneity of the Mobile Edge Computing (MEC) and its challenges, such as selection of proper application and programming models, accurate estimation of energy consumption, efficient management of simultaneous offloading, and standardization. Addressing following key challenges regarding computation offloading into the MEC:

- A decision on the computation offloading to the MEC with the purpose to determine whether the offloading is profitable for the UE.
- An efficient allocation of the computing resources within the MEC if the computation is offloaded in order to minimize execution delay and balance load.
- Mobility management for the applications offloaded to the MEC guaranteeing service continuity if the UEs exploiting the MEC roams throughout the network

B. Use Cases and Service Scenarios

- Consumer-Oriented Services: The first use case category is consumer-oriented and should be beneficial directly to the end-users. The computation offloading to the MEC enables running new emerging applications at the UEs, such as Web accelerated browsers, face/speech recognition, image/video editing, augmented, assisted, or virtual reality applications. Low latency applications, such as online gaming or remote desktop, may also benefit from the MEC in proximity.
- Operator and Third Party Services: The Mobile Edge Computing (MEC) is an IoT gateway that can be used to aggregate and deliver IoT services into highly distributed mobile base stations. It can also be used to extend the connected car cloud into the mobile network, allowing roadside applications to receive local messages and analyze them, and broadcast warnings to nearby vehicles with low latency. This was demonstrated by Nokia and its partners in an operator's LTE network.
- Network Performance and QoE Improvement Services: The third category of use cases are those optimizing network performance and/or improving QoE. An analytic application exploiting the MEC can provide real-time information on traffic requirements of the radio/backhaul network, while an optimization application can reshape the traffic per application or re-routes traffic as required. Local content caching at the mobile edge can help alleviate congested backhaul links, while radio network optimization can be used for mobile video delivery optimization using throughput guidance for TCP.

C. MEC Architecture and Standardization

MEC concepts are proposed to integrate cloud capabilities into mobile network architecture.

- Small Cell Cloud (SCC) - The SCC concept is to enhance small cells (SCeNBs) by providing additional computation and storage capabilities, using network function virtualization (NFV) and virtualization.
- Mobile Micro Clouds (MMC) - The MMC concept allows users to have instantaneous access to the cloud with low latency.
- Fast Moving Personal Cloud (MobiScud) - The MobiScud architecture integrates cloud services into mobile networks while maintaining backward compatibility with existing networks.
- Follow Me Cloud (FMC) - The FMC concept leverages the fact that mobile operators need to decentralize their networks to cope with the growing number of UEs, replacing the centralized CN with a distributed one.
- CONCERT - is a concept converging cloud and cellular systems, utilizing NFV principles and SDN technology.

ETSI is actively standardizing the Mobile Edge Computing (MEC) to integrate it into mobile networks.

- Standardization of ETSI MEC: ISG MEC has developed a PoC specification to promote the MEC, illustrate key aspects, and build confidence in its viability.
- ETSI MEC Reference Architecture: The reference architecture of ETSI MEC is composed of functional elements and reference points allowing interaction between them, with a user application lifecycle management (LCM) proxy and a mobile edge orchestrator.
- Deployment Options of ETSI MEC: MEC servers can be deployed at base stations, cell aggregation sites, or multi-RAT aggregation points, depending on scalability, physical deployment constraints, and performance criteria.

D. Introduction to computation offloading

A decision on computation offloading may result in: Local execution , Full offloading or Partial offloading An important aspect in the computation offloading is also an application model/type since it determines whether full or partial offloading is applicable, what could be offloaded, and how. we classify the applications according to several criteria like Offloadability of application, Knowledge on the amount of data to be processed, Dependency of the offloadable parts

E. Decision on computation offloading to MEC

- Full Offloading: The main objective of the works focused on the full offloading decision is to minimize an execution delay, to minimize energy consumption at the UE while predefined delay constraint is satisfied, or to find a proper trade-off between both the energy consumption and the execution delay.
- Partial Offloading: We classify the research on works focused on minimization of the energy consumption at the UE while predefined delay constraint is satisfied and works finding a proper trade-off between both the energy consumption and the execution delay.

Computation offloading strategies can achieve up to 90 percent energy savings and up to 98 percent reduction in execution delay.

F. Allocation of computing resources

A proper allocation of computing resources for applications offloaded to the MEC is determined by the ability of the offloaded application to be parallelized/partitioned.

- Allocation of Computation Resources at a Single Node: The authors propose a priority based cooperation policy to maximize the amount of applications processed in the MEC while satisfying their delay requirements. The objective of this paper is to minimize execution delay, power consumption, communication and computing resource overloading, and VM migration cost.
- Allocation of Computation Resources at Multiple Nodes (Federated Clouds): The main objective of the papers is to minimize execution delay and/or power consumption of computing nodes and balance communication and computing loads. The paper proposes three clustering strategies to minimize execution delay and power consumption of computing nodes. Joint clusters optimization is able to minimize the power consumption of clusters while

guaranteeing required execution delay for each UE. The Application Considering Algorithm (ACA) is proposed to balance communication and computation load of SCeNBs while satisfying the delay requirement of the offloaded application.

G. Mobility management for MEC

In conventional mobile cellular networks, a mobility of users is enabled by handover procedure when the UE changes the serving eNB/SCeNB as it roams throughout the network. If the UE offloads computation to the MEC, it is important to ensure service continuity and QoS. There are several options to cope with the mobility of UEs, such as adapting transmission power during the time when the offloaded application is processed by the MEC. Alternatively, service continuity can be guaranteed by VM migration (i.e., the process during which the VM run at the current computing node is migrated to another, more suitable, computing node) or selection of a new communication path between the UE and the computing node.

H. Conclusion

The paper summarises the lessons learnt from the state of the art focusing on computation offloading into the MEC. It additionally discusses several open research challenges not addressed by the current researchers like Distribution and Management of MEC Resources, Offloading Decision, Allocation of Computing Resources, Mobility Management, Traffic Paradigm Imposed by Coexistence of Offloaded Data and Conventional Data and Concept Validation.

REFERENCES

[1] Jian Shen , , Tianqi Zhou , Debiao He , Yuexin Zhang,(2019). Block Design-Based Key Agreement for Group Data Sharing in Cloud Computing https://ieeexplore.ieee.org/document/8543671

[2] Armstrong Nhlabatsi , Jin B. Hong , Dong Seong Kim, Rachael Fernandez, Alaa Hussein, Noora Fetais , and Khaled M. Khan (2021). Threat-Specific Security Risk Evaluation in the Cloud https://ieeexplore.ieee.org/document/8543671

[3] M. J. Hossain Faruk, S. Tahora, M. Tasnim, H. Shahriar and N. Sakib, "A Review of Quantum Cybersecurity: Threats, Risks and Opportunities," 2022 1st International Conference on AI in Cybersecurity (ICAIC), Victoria, TX, USA, 2022, pp. 1-8, doi: 10.1109/ICAIC53980.2022.9896970

[4] Y. Zhang, X. Lan, J. Ren and L. Cai, "Efficient Computing Resource Sharing for Mobile Edge-Cloud Computing Networks," in IEEE/ACM Transactions on Networking, vol. 28, no. 3, pp. 1227-1240, June 2020, doi: 10.1109/TNET.2020.2979807.

[5] W. Shi, J. Cao, Q. Zhang, Y. Li and L. Xu, "Edge Computing: Vision and Challenges," in IEEE Internet of Things Journal, vol. 3, no. 5, pp. 637-646, Oct. 2016, doi: 10.1109/JIOT.2016.2579198.

[6] P. Mach and Z. Becvar, "Mobile Edge Computing: A Survey on Architecture and Computation Offloading," in IEEE Communications Surveys Tutorials, vol. 19, no. 3, pp. 1628-1656, thirdquarter 2017, doi: 10.1109/COMST.2017.2682318.