Available online www.jsaer.com

***Journal of Scientific and Engineering Research*, 2023, 10(7):141-148**

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

Edge Computing and Content Delivery Networks: Bringing Data Processing Closer to Users

Nikhil Bhagat

Sr. Technical Account Manager – Network Specialist
Independent Scholar, Network Engineering
nikhil.bhagat90@gmail.com

Abstract Demand for faster data processing and optimized content delivery has fueled the adoption of edge computing and Content Delivery Networks (CDNs). Such technologies decentralize data storage by bringing processing and data closer to end users, decreasing latency and optimizing the network. Edge computing allows data to be processed at or close to the source, which is important for latency sensitive applications like IoT and autonomous systems. CDNs, on the other hand, optimize content distribution by storing data in geographically distributed servers, which improves loading times and minimizes the bandwidth load on origin servers. This paper provides an overview on the structure and function of edge computing and CDNs as well as how they can be utilized for a better user experience and scale. It also discusses the benefits of lower latency, optimized bandwidth, and cost savings offered by these technologies. At last, the most important design considerations, like security, load balancing, and cache design, are explored to provide an insight on deployment success. Edge computing and CDNs will be incredibly important for developing next-gen technologies and for determining the future of distributed networks in the digital age.

Keywords Edge Computing, Content Delivery Networks, Latency, Caching, best practices.

1. Introduction

The rapid growth of the Internet has brought new demands for fast, reliable and scalable processing of data. With the advent of cloud services, streaming platforms, and IoT devices, individuals expect instant access to data and content [1]. Traditional centralized compute architectures are usually incapable of meeting these needs due to a limited bandwidth, latency management and server capabilities. Two major advancements to meet such challenges are edge computing and Content Delivery Networks (CDNs) [2].

With edge computing, data processing and storage is localized near the point of use, reducing latency and bandwidth by moving workload away from centralized cloud servers [3]. CDNs, on the other hand, ensure faster delivery of content by storing copies of the data in multiple geographical regions to minimize the physical separation between user and server. Both technologies have the same goal of scaling and enhancing the performance and experience by decentralizing work and resources.

This paper aims to describe edge computing and CDNs, how they work, and the benefits of them. This paper will explore the design considerations involved in deploying these technologies within complex networks.

2. Evolution of Cloud Computing

Cloud computing has been a backbone of new IT infrastructure, which allows companies to replace their on-premises data centers with scalable, pay-as-you-go solutions [4]. Cloud computing, at its core, provides 3 services: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).

These applications leverage the massive centralized data centers run by clouds providers like Amazon Web Services (AWS), Microsoft Azure and Google Cloud, which enables users to virtualize computing tasks.

A. Advantages of Cloud Computing

• **Scalability:** Cloud infrastructure allows enterprises to elastically scale computing resources as needed without making costly capital investments to maintain data centers [5].

• **Cost Efficiency:** Cloud computing shifts Capital expenditure (CAPEX) model to an Operational expenditure (OPEX) model, allowing businesses to only pay for what they consume.

• **Global access:** Cloud Services provide global access for companies to easily deliver applications and services to any user with minimal overhead [6].

However, despite its revolutionary benefits, the inherent latency of centralized cloud data centers has proven incompatible with real-time use cases such as driverless cars, augmented reality and industrial IoT.

3. Edge Computing and Its Function

Edge computing is decentralized computing where the data is processed closer to its source instead relying on a centralized cloud environment. The computation in this model is done at or near the "edge" of the network i.e. on local devices, edge servers, gateways [7]. This model eliminates the need of routing data to central cloud data centers, which are far from the user.

The primary aim of edge computing is to make data processing more efficient by reducing the latency and optimize bandwidth consumption [8]. If the data is processed locally or near the user, the application will quickly respond to requests and it will be able to analyze and process the data in real time. This works especially well for latency sensitive applications like self-driving cars, IoT devices and AR [9].

Additionally, edge computing is highly scalable, allowing organizations to scale computing workloads across different locations. This takes the pressure off centralized cloud servers, while making it possible to manage traffic spikes or device connections without impacting performance.

4. Edge Computing Architecture

Edge computing architecture is useful for use cases involving real-time data processing, such as smart cities, autonomous vehicles, and industrial IoT applications. Edge computing architecture can be separated into several layers that do their respective tasks while contributing to the efficiency and scalability of the entire system.

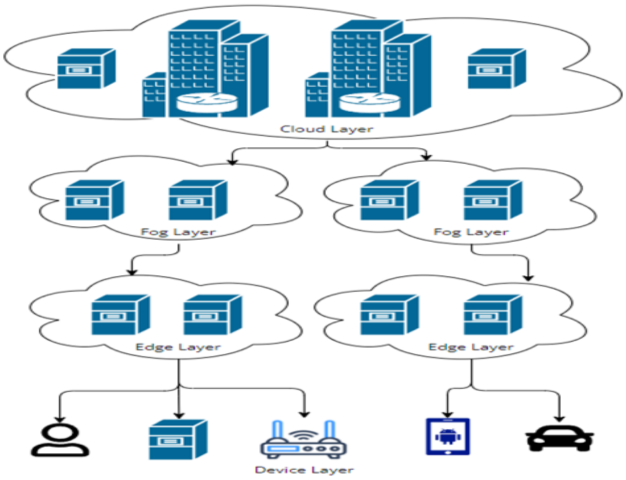

Fig. 1 *Edge Computing Architecture*

A. Device Layer

The device layer consists of end-user nodes or edge nodes like sensors, actuators, IoT devices, or phones [10]. These devices collect data and in some cases do light processing locally. For instance, a home's smart thermostat can read environmental data and adjust the temperature at home without passing everything to the cloud. The device layer is useful for applications that are latency sensitive as it helps eliminate the need for distant connectivity.

B. Edge Layer

The edge layer is where the bulk data processing takes place. This layer consists of edge servers, gateways, or micro-data centers near the data source. These edge nodes perform computation, storage and filtering of the data. Edge servers can collect data from a several devices, process locally, and send only the pertinent data to a central cloud server for processing or storage [11]. This layer takes huge load off of the network and cloud infrastructure as data isn't transported far to be processed.

Edge nodes are placed in a wide range of geographical areas, such as a factory, store, or smart city, allowing real-time responses. For instance, in a smart city, edge servers deployed at traffic intersections can process real-time traffic data to regulate signals, reducing latency and enabling faster decision-making.

C. Fog Layer (Optional)

Some edge computing solutions even include a fog layer, serving as a gateway between the edge devices and the cloud. Fog computing allocates the processing, storage and networking resources at various nodes into a tree structure [12]. This enables more fine-grained control and also reduces latency even more by handling data near the origin. This fog layer is particularly helpful in large scale IoT systems, where multiple intermediate levels of processing are required to mitigate latency and manage data flows.

D. Cloud Layer

The cloud layer remains vital to edge computing, particularly for long-term storage, advanced data analytics, and high-end processing operations involving large amounts of computing capacity [13]. The cloud serves as the backend to edge computing architectures that process data from edge nodes and provide auxiliary services including machine learning, big data analysis, and backup.

E. Communication Layer

The communication layer acts as the connector between layers in the edge computing design. It enables data transmission between end-user devices, edge nodes, and the cloud. The various routing protocols are selected as per the need of the system: Low Latency, High Bandwidth, Security etc. Standard communication technologies are Wi-Fi, 4G/5G, Ethernet, and wireless sensor networks (WSNs) [14]. Choice of the routing protocol depends on various factors including the size of the network, required performance, and network security.

F. Security and Management Layer

Security and management play a vital role in edge computing system. As data traverses through multiple layers, edge computing introduces security challenges such as data privacy, authentication, and secure communication [15]. The management layer keeps edge nodes up and running, enforces security policies, and takes care of software updates and patches across the distributed devices. A strong security layer is necessary for edge computing solutions like encryption, secure access control, and anomaly detection to guard against unauthorized access or attacks [16].

5. Content Delivery Networks and How They Operate

Content Delivery Networks (CDNs) are a cluster of distributed servers that are strategically placed across the world to better serve web content, including images, videos, and HTML files. CDNs help in streamlining the delivery process by keeping data locally cached near the user, reducing the need to access the origin server [17].

A CDN's core purpose is to deliver content as close as possible to a user, rather than sending it from a remote origin server. When a customer accesses content, the CDN routes it to the nearest edge server that has a copy of the data. This minimizes the round-trip towards the origin server, thus improving load times and latency.

Moreover, CDNs distribute network traffic across multiple servers, decreasing the chances of network bottlenecks and congestion [18]. This is particularly useful during high-volume events, such as live sports or concert streaming, where a sudden surge in demand can overwhelm origin servers.

6. Content Delivery Networks (CDN) Architecture

The primary goal of a CDN is to optimize, accelerate and improve the quality of content delivery (web pages, videos, images, and other digital content) by reducing the distance between the consumer and the content server. CDNs are commonly used to enhance user experience by reducing latency and ensuring content availability during periods of high demand. A CDN has a number of important pieces that collaborate to ensure the best possible distribution of content.

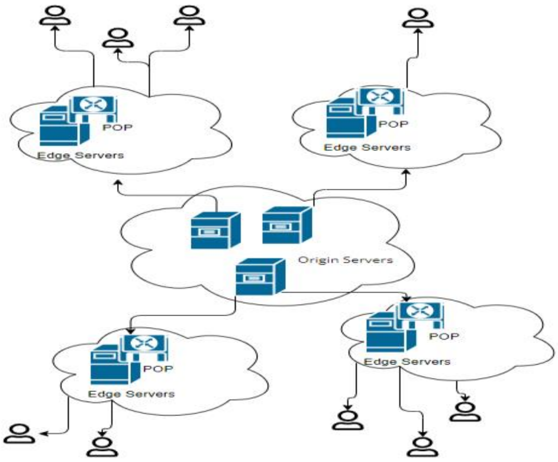

Fig. 2 *High Level Content Delivery Network Architecture*

A. Origin Servers

The origin server is the main server where the source pages, images, videos reside. This server is owned by the content owner, who updates the content [19]. Although the origin server stores the master data, the majority of users do not use it directly. Rather, the content is distributed over several edge servers in the CDN network. It reduces the pressure on the origin server, so that it's not throttled by a lot of requests during peak periods.

B. Edge Servers

Edge servers make up the core of a CDN. These servers are stacked closer to the users in geographically distributed datacenters known as Points of Presence (PoPs) [20]. When a user requests content, the CDN forwards the request to the nearest edge server. If the requested content is stored on the edge server, it is served right away reducing the distance between user and origin server. Without a cache the edge server fetches the content from origin server, caches it locally and provides it to the user [21]. Edge servers reduce latency and accelerate content distribution by minimizing the distance data needs to travel.

C. Points of Presence (PoPs)

Points of Presence (PoPs) are distributed data centers that house multiple edge servers. PoPs are strategically placed in high-traffic areas like big cities or densely populated places [22]. A CDN distributes cached content across multiple PoPs, making it available in real-time to users anywhere in the world. The closer a PoP is to a user, the faster the content can reach them. PoPs not only store information but also distribute the network load, which manages traffic surges and provides high availability.

D. Caching Mechanisms

Caching plays a vital role in a CDN design. CDN edge servers cache copy of frequently requested content for a short period of time. This decreases the number of times content needs to be pulled from the original server, decreasing bandwidth usage and enhancing the speed of delivery for the pages or media files [20]. CDNs use multiple caching rules such as time-to-live policies to enforce specific cache lifetime. This ensures that the cache server is continuously updated with new relevant data while minimizing server usage.

E. Load Balancing

CDN architecture includes load balancing. It helps distribute user requests over multiple edge servers in the network, without any one server being overwhelmed [23]. Load balancing engines route traffic to low latency or least busy edge server to reduce response time and optimize the CDN. Using load sharing, CDNs can scale to meet high volumes of traffic without compromising service quality.

F. CDN Controller

The CDN controller is the central management layer that manages the operations of the CDN. It takes care of sending requests to the right edge servers, keeping an eye on server health, and maintaining content updates across the network [24]. It also handles communication between the origin server and edge servers, ensuring consistent content and updates across all PoPs.

7. Advantages of Edge Computing and Content Delivery Networks

Edge computing and CDNs offer a multitude of benefits to companies and users. These are enabled because of their common vision to decentralize data processing and content sharing.

A. Low Latency

Whether processing information at the edge of the network or rendering close to users, both technologies minimize latency [11]. This is important for interactive systems, like online gaming, video calling and autonomous robots.

B. Enhanced Bandwidth Utilization

Edge computing reduces the amount of data required to be borne across the network and CDNs de-cache origin servers [11]. Collectively, they free up bandwidth, making networks able to carry more traffic with less capacity.

C. Improved User Experience

Rapider data and content delivery translate to better user experience. Websites load quicker, applications perform better, and the videos streams do not incur buffering issues.

D. Highly Scalable

Both edge computing and CDNs are scalable. Organizations can install more edge servers or CDN nodes if traffic spikes, keeping performance consistent as the demand increases [25].

E. High Availability and Reliability

CDNs provide redundancy, as they distribute content to multiple edge servers. In the same way, edge computing also reduces dependency on centralized servers, ensuring that the failure of one region in the network doesn't affect the entire network [26].

F. Cost Efficiency

As it puts a minimal strain on centralized servers, and avoids sending the data across long distances, edge computing and CDNs can save a significant amount of expenses in the long run especially for bandwidth intensive applications.

8. Design Considerations While Using Edge Computing or A CDN Solution

When designing an edge computing or CDN solution, it is important to take into account a few key points, ensuring optimal performance, scalability and security.

A. Latency requirements with Proximity to Users

One of the major reasons for use of edge computing and CDNs is to eliminate the latency through centralized data processing and content delivery near consumers [18]. For an edge computing/CDN solution, the geography of edge servers or PoPs needs to be thoughtfully planned. These servers can be placed around large user areas or high traffic regions to reduce latency. Proximity to users is particularly valuable in real-time applications, like online games, livestreaming, and IoT devices, where data latency can affect user experience and device performance [27].

B. Scalability and Load Balancing

Edge computing and CDN architectures should be scalable for higher workloads and high traffic volumes. Scalability means that your system can scale up as the data processing or content requests requirement increase, without a drop in the performance [28]. Load balancing ensures traffic is balanced across multiple edge servers (PoPs) to avoid a single node from being overwhelmed. An organization can achieve load balancing using various algorithms such as round-robin, least connections, and proximity routing to maintain high availability [29].

C. Caching Strategies

For CDNs, caching is essential for reducing bandwidth consumption and accelerating content load. The right caching policies like Time-to-live (TTL) configuration and cache purging ensures that high frequency content is cached on edge servers for faster delivery of frequently accessed content [23]. Caching policies should be dynamic and vary depending on the content delivered. Static files (such as images and videos) are cached for longer periods of time, while dynamic data (such as personalized information) may need to be updated more frequently.

D. Security Considerations

Security becomes an essential part of edge computing and CDN environments as the data is accessed and stored in different geographical locations. Edge nodes and PoPs can also be vulnerable to cyberattacks like DDoS attacks, data breaches and hacking. Using encryption for data transfers, rigorous access controls, and monitoring the network for anomalies are some of the best practices an organization must adopt. Moreover, CDNs can use DDoS protection such as rate limiting and traffic filtering to protect their network from malicious traffic [30].

E. Compliance and Privacy Protection

Edge computing usually entails running data at the edge of the network, near the source. In case of data-sensitive industries such as healthcare and financial services sectors, they might need their data to be protected by a strict privacy law such as the General Data Protection Regulation (GDPR) or the Health Insurance Portability and Accountability Act (HIPAA) [31]. If an organization is designing an edge solution, it's important to take various data privacy laws into consideration and carefully assess how data is processed, stored, and communicated in compliance with laws and regulations. Data encryption, anonymization, and local storage are some of the options organizations can use to ensure compliance [32].

F. Network Availability and Reliability

Both edge computing and CDN systems need high availability and reliability. Network, server or connectivity issues can disrupt services and negatively impact user experience [33]. To resolve this, the system needs to have redundancy in place. This can be achieved by having more than one edge server or PoP to act as a secondary backup solution in case the primary fails. Failover switches, backup servers, and redundant communications links can ensure high availability and service continuation.

G. Cost Considerations

Enterprises need to evaluate the deployment and operation cost of distributed infrastructure. These include hardware, software, data transfer, and ongoing maintenance costs [34]. This requires striking a proper balance between performance requirements and the price of the solution, so that the system scales without an undue operational burden.

9. Conclusion

Edge computing and Content Delivery Networks (CDNs) are one of the most innovative technologies in today's digital age, enabling data processing and content delivery at faster speeds. Since both technologies abstract computational tasks from the server and push content towards consumers, it overcomes the very underlying issues of traditional centralized approaches - latency, bandwidth and performance issues.

Edge computing removes physical boundaries between data sources and the computation points by performing computation closer to the user. This benefits latency sensitive applications such as IoT, driverless cars, and smart cities. Its architecture, which spans devices, edge servers, and origin servers, provides the ability to process data in real time, scale effectively, and optimize network performance.

CDNs, on the other hand, help distribute website content efficiently by caching data in strategically placed servers around the world. This shortens the distance between content and users, speeds up load times and minimizes network congestion. The CDN structure including origin servers, edge servers, PoPs, caching and load balancing, ensure high availability, reliability and better user experience during the peak hours.

Edge computing and CDNs are key to the future of distributed networks, particularly in industries embracing new technologies such as 5G, AI and the Internet of Things. These technologies enable innovation in various fields, from smart infrastructure to cloud gaming. Their enhanced performance, scalability, and reliability make them essential drivers of progress. When designing an edge computing or CDN solution, architects must consider several factors discussed in this paper to ensure all company requirements are addressed.

References

[1]. E. G. Renart, J. Diaz-Montes, and M. Parashar, "Data-Driven Stream Processing at the Edge," 2017 IEEE 1st International Conference on Fog and Edge Computing (ICFEC), May 2017, pp. 1-10. doi: 10.1109/icfec.2017.18.

[2]. Y. Zhao, W. Wang, Y. Li, C. Colman-Meixner, M. Tornatore, and J. Zhang, "Edge Computing and Networking: A Survey on Infrastructures and Applications," IEEE Access, vol. 7, pp. 101213-101230, Jan. 2019. doi: 10.1109/access.2019.2927538.

[3]. S. Douch, M. R. Abid, K. Zine-Dine, D. Bouzidi, and D. Benhaddou, "Edge Computing Technology Enablers: A Systematic Lecture Study," IEEE Access, vol. 10, pp. 69264-69302, Jan. 2022. doi: 10.1109/access.2022.3183634.

[4]. R. A. P. Rajan, "Evolution of Cloud Storage as Cloud Computing Infrastructure Service," IOSR Journal of Computer Engineering, vol. 1, no. 1, pp. 38-45, Jan. 2012. doi: 10.9790/0661-0113845.

[5]. Q. Zhang, C. Lü, and R. Boutaba, "Cloud computing: State-of-the-art and research challenges," Journal of Internet Services and Applications, vol. 1, no. 1, pp. 7-18, Apr. 2010. doi: 10.1007/s13174-010-0007-6.

[6]. J. Kozhipurath, "Cloud Service Costing Challenges," 2012 International Conference on Cloud Computing and Enterprise Management, pp. 1-6, Oct. 2012. doi: 10.1109/ccem.2012.6354604.

[7]. B. Varghese, N. Wang, S. Barbhuiya, P. Kilpatrick, and D. S. Nikolopoulos, "Challenges and Opportunities in Edge Computing," 2016 International Conference on Smart Cloud (SmartCloud), pp. 1-8, Nov. 2016. doi: 10.1109/smartcloud.2016.18.

[8]. M. Y. Uddin and S. Ahmad, "A Review on Edge to Cloud: Paradigm Shift from Large Data Centers to Small Centers of Data Everywhere," 2020 IEEE 3rd International Conference on Information and Communications Technology (ICICT), Feb. 2020, pp. 1-10. doi: 10.1109/icict48043.2020.9112457.

[9]. T. Kämäräinen, M. Siekkinen, A. Ylä-Jääski, W. Zhang, and P. Hui, "Dissecting the End-to-end Latency of Interactive Mobile Video Applications," Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services (MobiSys 2017), pp. 61-66, Feb. 2017. doi: 10.1145/3032970.3032985.

[10]. V. Karagiannis and S. Schulte, "Comparison of Alternative Architectures in Fog Computing," 2020 IEEE 4th International Conference on Fog and Edge Computing (ICFEC), pp. 1-10, May 2020. doi: 10.1109/icfec50348.2020.00010.

[11]. S. Singh, "Optimize Cloud Computations Using Edge Computing," 2017 International Conference on Big Data and Smart Computing (BID), Dec. 2017. doi: 10.1109/bid.2017.8336572.

[12]. D. M. A. D. Silva, G. Asaamoning, H. Orrillo, R. C. Sofia, and P. Mendes, "An Analysis of Fog Computing Data Placement Algorithms," 2019 International Conference on Distributed Computing Systems (ICDCS), pp. 1-10, Nov. 2019. doi: 10.1145/3360774.3368201.

[13]. A. Al-Qamash, I. Soliman, R. Abulibdeh, and M. Saleh, "Cloud, Fog, and Edge Computing: A Software Engineering Perspective," 2018 IEEE International Conference on Computer Applications (COMAPP), pp. 1-10, Aug. 2018. doi: 10.1109/comapp.2018.8460443.

[14]. H. B. Prasad and S. Babu, "A Survey on Network Routing Protocols in Internet of Things (IOT)," International Journal of Computer Applications, vol. 160, no. 2, pp. 18-22, Feb. 2017. doi: 10.5120/ijca2017912973.

[15]. X. Li, T. Chen, Q. Cheng, S. Ma, and J. Ma, "Smart Applications in Edge Computing: Overview on Authentication and Data Security," IEEE Internet of Things Journal, vol. 8, no. 6, pp. 4063-4080, Aug. 2020. doi: 10.1109/jiot.2020.3019297.

[16]. A. M. Alwakeel, "An Overview of Fog Computing and Edge Computing Security and Privacy Issues," Sensors, vol. 21, no. 24, pp. 8226-8226, Dec. 2021. doi: 10.3390/s21248226.

[17]. I. Lazăr and W. Terrill, "Exploring Content Delivery Networking," IEEE Internet Computing, vol. 3, no. 4, pp. 47-49, Jan. 2001. doi: 10.1109/6294.946620.

[18]. R. Krishnan, H. V. Madhyastha, S. Srinivasan, S. Jain, A. Krishnamurthy, T. E. Anderson, and J. Gao, "Moving Beyond End-to-end Path Information to Optimize CDN Performance," Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement (IMC 2009), pp. 1-10, Nov. 2009. doi: 10.1145/1644893.1644917.

[19]. A. Wan, A. Rigumye, and P. H. Jones, "Profile Based Routing and Billing Multimedia Content Delivery Network," Proceedings of the 21st International Conference on Computer Communications and Networks (ICCCN), May 2006, pp. 1-10. doi: 10.1109/icniconsmcl.2006.176.

[20]. P. Panchal, N. M. Ramaswamy, X. Su, and Y. Dong, "Content Delivery Networks in the Cloud with Distributed Edge Servers," 2013 IEEE International Conference on High Performance Computing and Communications (HPCC), Nov. 2013, pp. 1-10. doi: 10.1109/hpcc.and.euc.2013.81.

[21]. S. Roy, J. Ankcorn, and S. Wee, "Architecture of a Modular Streaming Media Server for Content Delivery Networks," IEEE International Conference on Multimedia and Expo (ICME), Jan. 2003, pp. III-569-III-570. doi: 10.1109/icme.2003.1221375.

[22]. M. Zhanikeev, "Fog Caching and a Trace-Based Analysis of Its Offload Effect," International Journal of Interdisciplinary Telecommunications and Networking (IJITN), vol. 10, no. 2, pp. 50-68, Feb. 2017. doi: 10.4018/ijitsa.2017070104.

[23]. E. Xinhua and B. Zhu, "A Distributed Cache Update Deployment Strategy in CDN," Journal of Physics: Conference Series, vol. 1004, Apr. 2018, pp. 012031-012031. doi: 10.1088/1742-6596/1004/1/012031.

[24]. D. Sarkar, N. Rakesh, and K. K. Mishra, "Content Delivery Networks: Insights and Recent Advancement," Proceedings of the 3rd IEEE International Conference on Parallel, Distributed and Grid Computing (PDGC 2016), Jan. 2016, pp. 1-5. doi: 10.1109/pdgc.2016.7913113.

[25]. T. L. Duc and P. Oestberg, "Application, Workload, and Infrastructure Models for Virtualized Content Delivery Networks Deployed in Edge Computing Environments," 2018 IEEE International Conference on Computer Communications and Networks (ICCCN), Jul. 2018, pp. 1-10. doi: 10.1109/icccn.2018.8487450.

[26]. P. T. Endo, M. Rodrigues, G. E. Gonçalves, J. Kelner, D. Sadok, and C. Curescu, "High Availability in Clouds: Systematic Review and Research Challenges," Journal of Cloud Computing, vol. 5, no. 1, Oct. 2016. doi: 10.1186/s13677-016-0066-8.

[27]. M. Claypool and D. Finkel, "The Effects of Latency on Player Performance in Cloud-Based Games," Proceedings of the 13th IEEE International Workshop on Network and Systems Support for Games (NetGames 2014), Dec. 2014. doi: 10.1109/netgames.2014.7008964.

[28]. M. M. Falatah and O. Batarfi, "Cloud Scalability Considerations," International Journal of Computer Science and Engineering Survey, vol. 5, no. 4, pp. 37-47, Aug. 2014. doi: 10.5121/ijcses.2014.5403.

[29]. H. Pydi and G. N. Iyer, "Analytical Review and Study on Load Balancing in Edge Computing Platform," 2020 International Conference on Computer Methods and Communication (ICCMC), Mar. 2020, pp. 1-6. doi: 10.1109/iccmc48092.2020.iccmc-00036.

[30]. K. Kalkan, G. Gür, and F. Alagöz, "Filtering-Based Defense Mechanisms Against DDoS Attacks: A Survey," IEEE Systems Journal, vol. 11, no. 4, pp. 2761-2773, Sept. 2016. doi: 10.1109/jsyst.2016.2602848.

[31]. H. K. Patil and R. Seshadri, "Big Data Security and Privacy Issues in Healthcare," 2014 IEEE International Congress on Big Data, Jun. 2014. doi: 10.1109/bigdata.congress.2014.112.

[32]. K. Scarfone, M. Souppaya, and M. Sexton, "Guide to Storage Encryption Technologies for End User Devices," NIST Special Publication 800-111, Jan. 2007. doi: 10.6028/nist.sp.800-111.

[33]. G. Semaan, "Designing Networks with the Optimal Availability," 2008 Optical Fiber Communication Conference (OFC), Feb. 2008, pp. 1-6. doi: 10.1109/ofc.2008.4528163.

[34]. A. Simonet, A. Lèbre, and A. Orgerie, "Deploying Distributed Cloud Infrastructures: Who and at What Cost?," Proceedings of the 3rd IEEE International Conference on Cloud Engineering (IC2E 2016), Apr. 2016, pp. 178-183. doi: 10.1109/ic2ew.2016.48.