Feature Article

Understanding Network Requirements for Smart City Applications: Challenges and Solutions

Nausheen Shoaib
National University of Computer and Emerging Sciences

Jawwad A. Shamsi
National University of Computer and Emerging Sciences

***Abstract*—Smart cities are contemporary revolutions that can handle the complexities of growing urban density. Smart applications reside in the cloud datacenter, where Internet of Everything devices or sources access these applications to obtain city services. Accessing applications from the distant cloud implies higher latency, huge network traffic, and possibilities of security and privacy breaches. These issues are not sustainable for real-time applications. This article describes network requirements and challenges faced by applications and solutions to encounter them.**

A SMART CITY utilizes urban informatics and technologies for providing city services on a large scale. It offers improved quality of life and a variety of innovative services such as energy, transport, healthcare, etc. Different smart applications continuously generate data from heterogeneous sources, such as mobile sensing devices and online social networks, including Internet of People (IoP) and wireless sensor networks. These applications are hosted at cloud-based datacenters.

Digital Object Identifier 10.1109/MITP.2018.2883047
Date of current version 21 May 2019.

Smart cities have complex network infrastructure comprising various networking devices at core, edge, and sensing layers. Network traffic (packets) from a diverse set of applications share network resources to access the distant cloud. These resources include communication links, switches, and network middle boxes. Therefore, accessing cloud applications leads toward higher communication latency, increased packet loss, and lower available bandwidth.

Smart city networks also suffer from security and privacy[1] violations through attacks, such as side channel, man in the middle, botnets, and cold boot attack.[2] Without sufficient security and protection mechanisms, users may resist

Published by the IEEE Computer Society

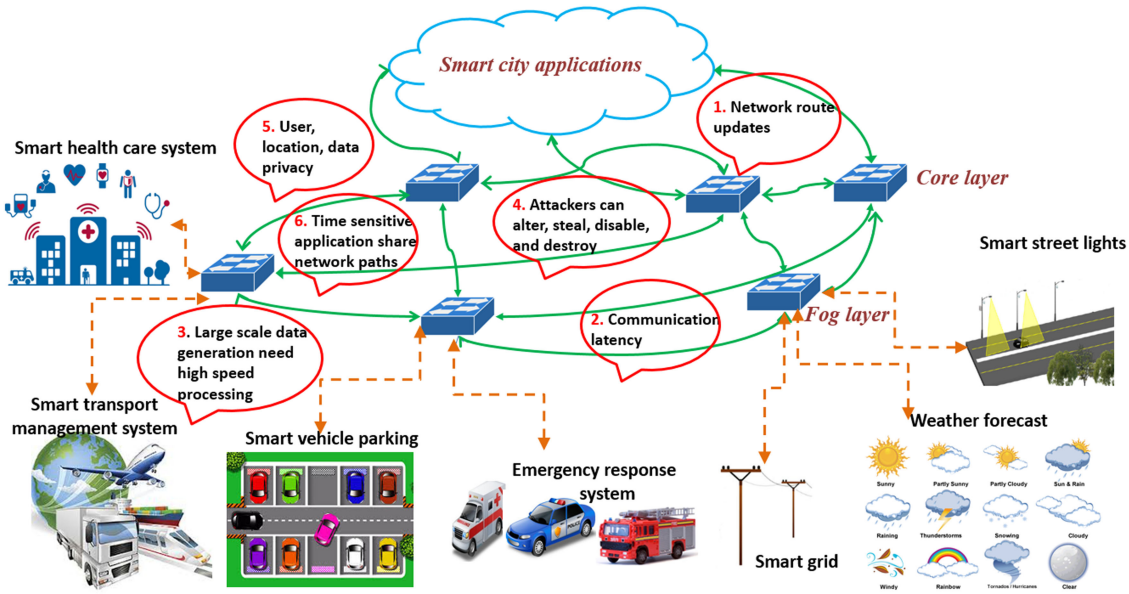

Figure 1. Requirements and challenges for network in a smart city.

accepting the smart city concept. Smart city applications also require high-speed network packet (header + payload) processing for quick response. Packet processing information is used for payload-based functions, such as malicious pattern detection and content based routing.

Further, different applications have variable network requirements in terms of response time and security. For instance, an emergency response system has to be agile, extremely secure, and time sensitive in response to a chaotic situation. Comparatively, a smart health application encompasses more flexible and relaxed requirements of these attributes. Since applications possess variable network requirements, a comprehensive solution is required to address these attributes.

This article analyzes application-specific network requirements and challenges. We also discuss the existing solutions and propose enhancements for efficient city services.

NETWORK REQUIREMENTS AND CHALLENGES

The high prospects and capabilities of the smart city lead us to explore an application's network needs. This section highlights the incorporation of various attributes, such as network programmability, packet processing, security, privacy, and quick response time. Figure 1 depicts the challenges through callouts.

Flexibility and Programmability

Millions of heterogeneous devices generate voluminous network traffic to access different applications in the cloud. This network traffic follows various route paths to access these applications. A city network must also comply with routing policies, which includes route computation and notification for changes. Additionally, the network must also conform to certain security rules for protection of assets from cyber threats. In this context, an efficient mechanism is required for seamless operation of city services.

One possible solution is to modify network devices individually and improve networking capability. However, this approach may increase cost and time. This leads to a substantial need for a programmable and a centralized network manager to apply network flow rules to various devices.

Communication Latency

Traditional resource management in a smart city relies on cloud facilities. These solutions are beneficial due to their unlimited computing capacity, cost efficiency, and elasticity. However, moving all data and services to the cloud adds several inconveniences to this option, such

Table 1. Smart applications classification based on security and timeliness.

S.No.	Smart city applications	Application components	Impact of security breach	Delay tolerance
1	Smart street lighting	***Remote monitoring***: lighting failures are reported to system.[3]	***Catastrophic:*** attacks in data can cause road accidents.	No
		Smart dimming: lights can be dimmed to conserve energy.[3]	***Moderate:*** street lights should not be dimmed in peak traffic hours	Yes
2	Smart grid	***Energy storage system:*** energy can be used in peak hours.[3]	***Critical:*** attacks can waste energy resources	No
		Load forecasting and monitoring of energy resources.[3]	***Catastrophic:*** attacks can result in failure of grid system	No
3	Smart traffic management	***Analytics:*** peak hour and direction of traffic flow.[3]	***Critical:*** attacks can miscalculate peak hour and inadequate traffic can cause severe blockage	No
		Analytics: distribution of traffic volume.[3]	***Catastrophic:*** attack can cause traffic jam or accidents	No
		Analytics: *estimation* of traffic growth rate.[3]	***Critical:*** attack can cause traffic blockage	No
		Analytics: alternative route computation.[3]	***Critical:*** attacks can divert traffic to wrong destinations	No
4	Smart health	Emergency response system.[3]	***Catastrophic:*** attack can cause severe issues with patient	No
		Electronic health record.[3]	***Catastrophic:*** modification attacks can cause severe issues with patient	No
5	Weather, news updates	Weather, news information	***Moderate:*** modification can be tolerable.	Yes

as high communication latency, risk of failure, and security vulnerabilities.

High-Speed Network Traffic (Packet) Processing

Smart sensing devices continuously generate large-scale network traffic (packet) to access cloud applications. These network packets pass through various network operations before accessing the cloud. Packet processing involves parsing of each packet and extracting useful information, such as the packet header and payload. Many network functions (NFs), such as content-based routing, malware detection, and firewall rules, use this information for further processing. Generally, CPU-based packet processing methods suffer from performance limitations due to their hardware design. Therefore, they need an effective solution to achieve high-speed packet processing.

Network Security

Protection and security is of utmost importance for smart city networks. Many smart applications share the city network, which makes them vulnerable to security-related issues, such as eavesdropping, tampering, and modification. Further, they are also susceptible to attacks, such as Sybil, Man in the Middle, and DoS attack. These factors may cause interruption and degradation in city services. The impact of these issues varies with each application. For instance, service interruptions in the smart grid may cause unbearable impact over end users. Comparatively, a waste management system can tolerate interruption of services. We have summarized the impact of network security on city services in Table 1. This significantly necessitates incorporation of a comprehensive network security mechanism.

Privacy

Data become a valuable asset, which can radically improve city services. Online social networks such as IoP and mobile crowd sensing[4,5] generate heterogeneous data, which suffers from various type of attacks such as Sybil attack, man in the middle attack, etc. These possible attacks lean toward user privacy issues, such as an individual's identity theft (name, address, phone number, etc.), leakage of sensitive data (occupation, health status, etc.), inferring sensitive information (smart meter reading to violate residence privacy), and violation of location privacy. Therefore, an effective solution is needed to ensure privacy issues.

Application-Specific Response Time

Massive network traffic accesses the distant cloud through a complex network infrastructure shared by many applications. Each application has variable response times to meet timeliness according to the city services provided by it. For instance, an emergency response system, such as an ambulance system, is a delay-sensitive application because it stringently needs to provide an immediate response. Comparatively, a smart parking system can tolerate some delay. Considering the time-sensitivity factor, it requires development of an efficient routing mechanism, which can result in better response time. Table 1 highlights timeliness requirements for different smart applications.

All of the above requirements and challenges encourage us to design a comprehensive solution to address the above needs.

PROPOSED SOLUTIONS

This section enlightens the solutions to fulfill network requirements and challenges. We propose a graphics processing units (GPUs) based multipurpose switch named GSwitch, which is composed of GPU-based network function virtualization (NFV). These NFVs are used for packet parsing, host- and network-based intrusion detection systems (HIDS/NIDS), and application classification (AC) for application-aware forwarding (AAF) mechanisms. We also recommend implementing privacy measures to overcome information leakage issues.

Network Programmability and Adaptability

The first requirement states the need of programmable and adaptable network services for smart applications. A software-defined network (SDN) ensures the incorporation of the above important components. The basic feature of SDN supports implementation of a centralized network manager, known as a SDN controller.[6] Network administrators can adapt the configuration on an SDN controller according to the current network status. We recommend the implementation of a controller at the cloud layer to obtain a global view of network paths. The controller communicates network policies such as route computation, updates, and selection of priority routes to all network devices.

Fog Layer

The second requirement focuses on reducing communication latency and security risks when communicating with the cloud layer. To circumvent this challenge, we propose to utilize fog computing,[7,8] which provides an intermediary layer between the sensing layer and a distant cloud. Implementing application-processing techniques at the fog layer reduces communication latency and response time. We recommend implementing the GSwitch at this layer, which is beneficial in reducing security risks and determining the preferred network route.

Accelerating Network Traffic (Packet) Processing

The third requirement imposes the need for high-speed packet processing[9,10] in order to achieve a fast response time. The emergence of NFV enables network applications to run on commodity hardware for flexibility and scalability. We propose to implement NFs such as packet parsing, malicious content detection, and application-aware forwarding at fog nodes. This implementation serves multiple purposes. First, it conserves network bandwidth as packets can be filtered before entering the core network. Further, it promotes cost effectiveness and programmability—the two incumbent requirements for serving a huge number of applications in a smart city. To accomplish high speed, we are inspired to use GPUNFVs.[11] Fog nodes can be equipped with multiple GPUs to promote scalability.

GPUs enhance packet-processing capabilities by exploiting thread-level parallelism. Advance GPU programming techniques can be used to achieve high-speed. These techniques permit sending network packets arbitrarily from CPU to GPU, or collectively in a batch.[12] The former method reduces the preprocessing delay; however, it induces an overhead of individually transferring each packet from CPU to GPU. Data transfer from CPU to GPU is considered as a bottleneck. In comparison, the latter technique reduces the overhead of sending each packet to the GPU as packets are transferred in a batch (group). This technique is appropriate for sending massive number of network packets to GPU device for further processing of other NFs, such as deep packet inspection (DPI), etc.

Network Security

The fourth requirement compels the importance of network security aspects for smart applications. We recommend implementing GPU-based HIDS by collecting and analyzing data about system calls, system events, and file systems. We also endorse incorporating GPU-based NIDS to prevent against various attacks, such as DoS attack, port scan, etc.

Generally, NIDS utilizes the information extracted from the DPI module and assists in detecting malicious activity in various ways. It detects malicious content in network packets and identifies suspicious activity by building a profile of routine network patterns that do not align with normal behavior. Further, NIDS also guards the network through firewall rules such that network traffic with dubious behavior is not permitted to enter the network.

In NIDS, pattern-matching operations are computationally extensive tasks and need fast execution. This technique requires a stream of packets compared with thousands of known virus signatures in a short amount of time. Several pattern matching algorithms[13,14] are in practice, such as Aho-Corasick or Rabin-Karp. Similarly, anomaly based detection methods entail network profiles for malicious and benign activities. This involves development of complex machine learning based models for training and detection. In the same context, the firewall uses packet headers and contents extracted from the DPI module[15] to safeguard the network using a set of rules.

In a smart city, security requirement varies with respect to each application. Our solution is to implement the AC module, which categorizes these applications into three different classes, i.e., catastrophic, critical, and moderate. Smart applications are divided into categories according to the impact of service interruption on city services. For instance, a catastrophic application such as smart grid, upon interruption, may have crucial effects. In comparison, the effect of interruption is reduced for critical and moderate applications. The classification module utilizes the information extracted from the DPI module and classifies applications according to their timeliness requirements. Application categories can be used to incorporate application-specific security attributes and route preferences for packet (data) forwarding to cloud datacenters. Table 1 elaborates AC based on their impact in executing fundamental operations of the city.

Privacy Measures

The fifth requirement is related to preserving user privacy during data sensing. A smart city network must incorporate privacy laws[1] as preventive measures. Further, security and privacy mechanisms such as encryption, anonymity, and access control[16] must be incorporated. Similarly, there are several authentication mechanisms, such as face identification and finger printing methods, that exist. However, these identification methods do not seem practical and convenient for end users, as it is difficult to authenticate users before offering city services. For this purpose, anonymous authentication methods such as pseudonyms, group signatures, and k-anonymity[16] can be implemented, which enables the authenticity without disclosure.

Application-Specific Path Selection

The sixth requirement emphasizes meeting application-specific timeliness requirements. For this purpose, preferred network route paths are computed by the AAF[17,18] mechanism. This GPUNFV uses the information extracted from DPI and AC modules. The application header consists of predefined tags useful in identifying

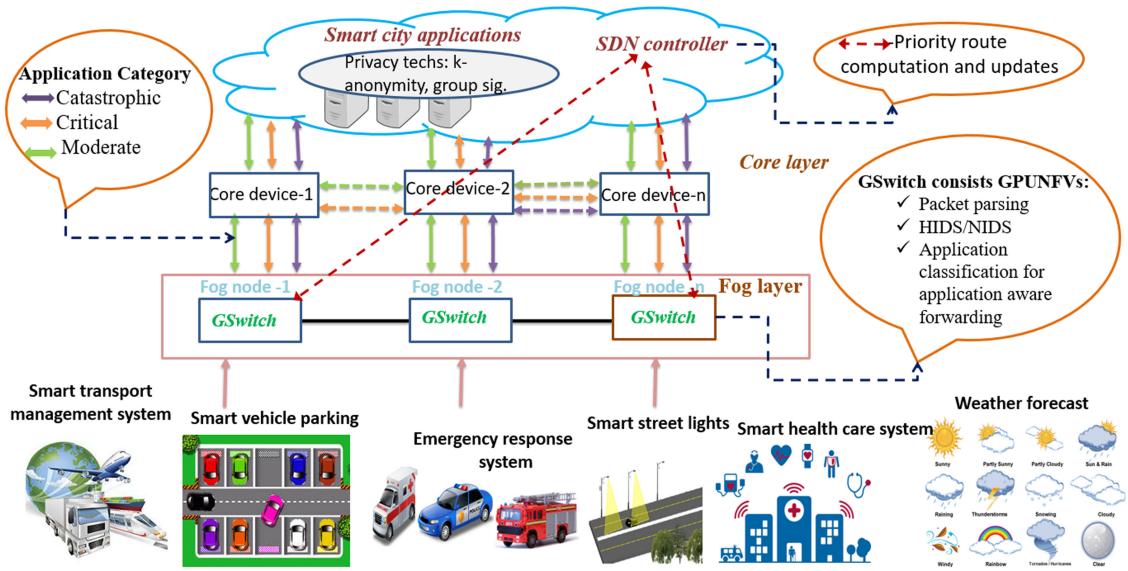

Figure 2. Proposed network framework for a smart city.

application class. Based on these classifications, network paths are computed by the SDN controller. They calculate network paths based on their estimated latency to the end user. In most cases, it is anticipated that the information can be transmitted directly from the fog node and the end user, thereby meeting the requirement of timeliness. For quick response, network paths are cached[19] by this module in GSwitch. They are also tags redundant paths to provide resilience against network failures.

METHODOLGY

This section explains the overall execution of our proposed framework, illustrated in Figure 2. Massive data are generated by various sources at the sensing layer to access cloud-based applications. Data are collected and aggregated at the fog layer by various fog nodes at distributed places.

GSwitch is incorporated at the fog layer, which reduces communication latency between end user and cloud. This framework serves multiple benefits in terms of latency and security and conserves network bandwidth.

In GSwitch, packet streams are sent to the GPU device to achieve high-speed processing. Packet parsing module parses the packet to extract the header and payload. The packet payloads are then passed from inspection mechanism for malicious content detection through GPU-based NIDS. These systems filter the incoming network packet (data) and forward the legitimate traffic (packet) to the cloud, therefore reducing network traffic and conserving network bandwidth. These security systems also prevent the network from intruders and attackers through well-defined security policies.

Moreover, the filtered network packets are received by the AC module. The classification is based on predefined tags using a machine-learning approach. Applications are divided into three broad categories, i.e., catastrophic, critical, and moderate, to achieve the timeliness requirement. The classified network packets are sent to the AAF module, which provides preferred routes paths to time-sensitive applications, which conserves network bandwidth and provides quick response. The AAF module coordinates with a cloud-based SDN controller for network flow rules, such as route computation, allocating priority routes. The network flow rules are cached at GSwitch, thus reducing the overhead of communicating the controller for each new packet. In the case of change in flow rules, the controller propagates them to GSwitch.

Since cloud-based smart applications also suffer from privacy violations, privacy laws and mechanisms such as k-anonymity and group signatures can be incorporated.

OPEN ISSUES

We anticipate that the implementation of the proposed network framework still has a few

open issues. We enlighten these concerns and their possible solutions.

Management of Core Network

A smart city network is shared by many applications. There are numerous inconveniences, such as control, management, maintenance, and administration, that need appropriate monitoring services for day-to-day operations. A possible solution is to manage the network through a consortium of major stakeholders, including government, ISPs, and users. The role of computer scientist is also important in designing and outlining network policies for network management while ensuring security and privacy.

Smart City App Store

Smart city services are extensively growing and demanding a rapid application development. These applications may be developed either through a consortium or by third parties. In both cases, consortium needs to monitor and federate the "Smart City App Store."

Application Level Programming

Application level programming still has concerns regarding packet size, format, and validation of the application header. Further research is required to determine the efficient size of the packet header.

Scalability and Fault Tolerance

Network scalability is also an important aspect. The integrated SDN controller may experience performance degradation with an increase in network traffic. One of the possible solutions is to incorporate a layer of the distributed controller. Consistency and load balancing in the distributed controller is still a main concern in a smart city.

Security and Vulnerability

Security and vulnerability is expected to grow. We provide solutions to the best of our knowledge; however, other security concerns are still there. These include network security at control and data planes, user access control, prevention against message fabrication, and mitigation of attacks. For this purpose, strict security mechanisms[20] can be considered.

CONCLUSION

The network requirement for smart cities is complex. Various concepts, applications, and technologies interact to encompass every aspect of an inhabitant's life. Therefore, designing an effective network framework, which is capable of providing efficient and secure city services through smart application, is a challenging task.

We have introduced taxonomies for flexible network programming: GPUNFVs for fast network traffic processing. Further, we also proposed application-specific route computation and its impact on security. We also recommend incorporating privacy laws and measures for prevention purposes. These taxonomies allowed us to present a holistic analysis of application's network needs and their possible solutions.

Either city government or consortium can implement our proposed framework. The role of computer scientist is important in nurturing an effective system, which has the capability to fulfill these requirements for the end user. We anticipate that such systems can grow with the emergence of new applications. In addition, we have also discussed some open issues for future research.

ACKNOWLEDGMENT

This work was supported in part by an NVIDIA Teaching and Research Center Grant and in part by the Higher Education Commission Pakistan under Grant NRPU-5946.

REFERENCES

1. J. A. Shamsi and M. A. Khojaye, "Understanding privacy violations in big data systems," *IT Professional*, vol. 20, no. 3, pp. 73–81, 2018.
2. K. Zhang *et al.*, "Security and privacy in smart city applications: Challenges and solutions," *IEEE Commun. Mag.*, vol. 55, no. 1, pp. 122–129, Jan. 2017.
3. Gharaibeh *et al.*, "Smart cities: A survey on data management, security, and enabling technologies," *IEEE Commun. Surveys Tut.*, vol. 19, no. 4, pp. 2456–2501, Oct.–Dec. 2017.
4. J. Wang *et al.*, "Social-Network-Assisted Worker Recruitment in Mobile Crowd Sensing," arXiv:1805.08525, 22 May, 2018.

5. J. Wang *et al.*, "Task allocation in mobile crowd sensing: State of the art and future opportunities," vol. 5, no. 5, pp. 3747–3757, Oct. 2018.
6. S. Chakrabarty and D.W. Engels, "A secure IoT architecture for smart cities," in *Proc. 13th IEEE Annu. Conf. Consumer Commun. Netw. Conf.,* Las Vegas, NV, USA, Jan. 2016, pp. 812–813.
7. A. C. Baktir, A. Ozgovde, and C. Ersoy, "How can edge computing benefit from software-defined networking: A survey, use cases, and future directions," *IEEE Commun. Surveys Tut.*, vol. 19, no. 4, pp. 2359–2391, Oct.–Dec. 2017.
8. K. Xue *et al.*, "Fog-aided verifiable privacy preserving access control for latency-sensitive data sharing in vehicular cloud computing," *IEEE Netw.*, vol. 32, no. 3, pp. 7–13, May/Jun. 2018.
9. Y. Go *et al.*, "APUNet: Revitalizing GPU as packet processing accelerator," in *Proc. 14th USENIX Symp. Netw. Syst. Des. Implementation*, Boston, MA, USA, Mar. 2017, pp. 83–96.
10. D. Cerovic *et al.*, "Fast packet processing: A survey," *IEEE Commun. Surveys Tut.*, vol. 20, no. 4, pp. 3645–3676, Oct.–Dec. 2018.
11. X. Yi, J. Duan, and C. Wu, "GPUNFV: A GPU-accelerated NFV system," in *Proc. 1st Asia-Pacific Workshop Netw.*, Aug. 2017, pp. 85–91.
12. Z. Zheng *et al.*, "BLOP: Batch-level order preserving for GPU-accelerated packet processing," in *Proc. ACM SIGCOMM Posters Demos*, 2017, pp. 136–137.
13. C. Xu *et al.*, "A survey on regular expression matching for deep packet inspection: Applications, algorithms, and hardware platforms," *IEEE Commun. Surveys Tut.*, vol. 18, no. 4, pp. 2991–3029, Oct.–Dec. 2016.
14. C. L. Hung, C. Y. Lin, and P. C. Wu, "An efficient GPU-based multiple pattern matching algorithm for packet filtering," *J. Signal Process. Syst.*, vol. 86, no. 2/3, pp. 347–358, 2017.
15. N. Shoaib *et al.*, "GDPI: Signature based deep packet inspection using GPUs," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 11, pp. 210–216, 2017.
16. J. Ni *et al.*, "Securing fog computing for internet of things applications: Challenges and solutions," *IEEE Commun. Surveys Tut.*, vol. 20 no. 1, pp. 601–628, Jan.–Mar. 2017.
17. U. Pongsakorn *et al.*, "Application-aware network: Network route management using SDN based on application characteristics," *CSI Trans. ICT*, vol. 5, no. 4, pp. 375–385, 2017.
18. S. T. Pasca, S. S. Kodali, and K. Kataoka, "AMPS: Application aware multipath flow routing using machine learning in SDN," in *Proc. 23rd IEEE Nat. Conf.Commun.*, Mar. 2017, pp. 1–6.
19. L. Cui, F. R. Yu, and Q. Yan, "When big data meets software-defined networking: SDN for big data and big data for SDN," *IEEE Netw.*, vol. 30, no. 1, pp. 58–65, Jan./Feb. 2016.
20. J. A. Shamsi, S. Zeadally, and Z. Nasir, "Interventions in cyberspace: Status and trends," *IT Professional*, vol. 18, no. 1, pp. 18–25, 2016.

Nausheen Shoaib is a lecturer with the National University of Computer and Emerging Sciences (NUCES), Karachi, Pakistan. She is currently working toward the Ph.D. degree in computer science at NUCES. She has been a Microsoft certified professional. Her research interests include software-defined networks, network security, and high-performance computing. She received the B.S. degree in electronics engineering from Sir Syed University and the M.S. degree in computer communications and networks from Hamdard University, Karachi, in 2013. Contact her at nausheen.shoaib@nu.edu.pk.

Jawwad A. Shamsi is a professor of computer science and the director of campus at the National University of Computer and Emerging Sciences (NUCES), Karachi, Pakistan. His work has been funded by the Higher Education Commission, ICTRDF, Pakistan, and NVIDIA, USA. He received the IEEE TCPP early adopter awards in 2012, 2013, and 2015. He served as a head of the CS Department at NUCES. His research interest lies in high-performance computing, software-defined networking, and network security. He received the Ph.D. degree in computer science from Wayne State University, Detroit, MI, USA, in 2009. Contact him at jawwad.shamsi@nu.edu.pk.