International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 04 Issue 14
November 2017

Survey on Edge Cloud and Edge Computing with Emerging Technologies

[1]VASAVI SRAVANTHI BALUSA, [2]M.THANMAYEE

[1,2] Assistant Professor, Department of Computer Science and Engineering,
TKR College of Engineering and Technology, Hyderabad, Telangana.

Abstract: - *The Internet is developing quickly toward the future "Internet of Things" (IoT) which will possibly connect billions or even trillions of edge devices which could generate massive amount of data at a very high speed and some of the applications may require very low latency. The traditional cloud infrastructure will run into a series of difficulties due to centralized computation, storage, and networking in a small number of datacenters, and due to the relative long distance between the edge devices and the remote datacenters. To attack this challenge, edge cloud and edge computing seem to be a promising possibility which provides resources closer to the resource-poor edge IoT devices and potentially can nurture a new IoT innovation ecosystem. Such prospect is enabled by a series of emerging technologies including Network Function Virtualization (NFV) and Software Defined Networking (SDN). In this survey paper, we examine characteristics among traditional and edge cloud and edge computing with Edge computing security.*

Keywords: *Internet of Things, edge cloud, edge computing, Network Function Virtualization (NFV), Software Defined Networking (SDN)*

1. Introduction

The conventionally centralized cloud computing model favors several large-sized distributed datacenters. It has proved to be a huge success in the current Internet and was broadly adopted by the aforementioned giant corporations. The success can be attributed to several factors:

(1) It provides an on-demand pay-as-you- go service to the users which lowers the owning cost for general customers;

(2) It provides elasticity of computing, storage, and networking resources which is flexible and scalable;

(3) it facilitates big-data analytics using machine learning technologies due to the highly centralized colocation of intensive computation and data. In short, it is through economics of scale in operations and system administration that the conventional cloud computing wins.

However, such a centralized model will face significant challenges toward the IoT world and we briefly discuss some.

(1) Volume and velocity of data accumulation of IoT devices. In current model, the new application delivery highly depends on giant companies' proprietary overlays and tools, and they generally have to transfer all the data from the edge devices to the remote datacenters, which will not be possible considering the volume and velocity of the data generated by the IoT devices in the future.

(2) Latency due to the distance between edge IoT devices and datacenters. The centralized cloud model also leads to a fact that the edge devices (often mobile) are usually relatively far away from the datacenters. In the future when the number of edge devices experiences exponential increase, it is imaginable that high latency can be a big challenge for quite a number of applications that involve end-to-end communications.

(3) Monopoly vs. open IoT competition. Current centralized cloud infrastructure is usually expensive to build and is only affordable to those giant companies that tend to define and use proprietary protocols. Customers are easily stuck to some specific infrastructures as the cost of switching to others could be dreadful. Such lack of openness could lead to a monopoly, ossification of the Internet, and further inhibit innovations.

In short, we need to address the deficiencies of the traditional cloud computing model. In our opinion, an open edge cloud infrastructure is inevitable and necessary to embrace the paradigm shift to the future IoT world.

What exactly is edge computing?

Edge computing is a "mesh network of micro data centers that process or store critical data locally and push all received data to a central data center or cloud storage repository, in a footprint of less than 100 square feet," according to research firm IDC.

It is typically referred to in IoT use cases, where edge devices would collect data – sometimes massive amounts of it – and send it all to a data center or cloud for processing. Edge computing triages the data locally so some of it is processed locally, reducing the backhaul traffic to the central repository.

Typically, this is done by the IoT devices transferring the data to a local device that includes compute, storage and network connectivity in a small form factor. Data is processed at the edge, and all or a portion of it is sent to the central processing or storage repository in a corporate data center, co-location facility or IaaS cloud.

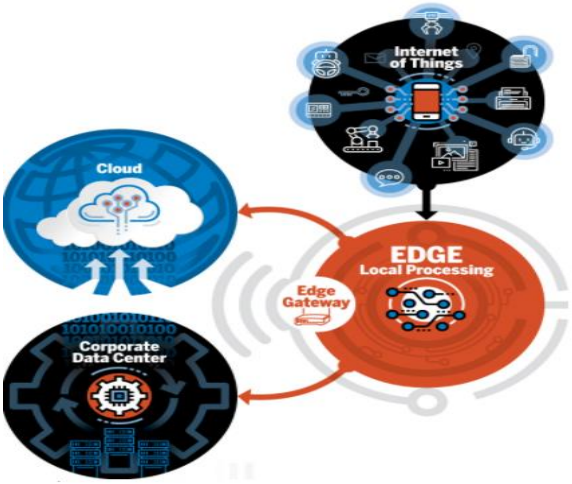


Figure 1. Edge computing work flow

Edge cloud infrastructures

With open edge cloud infrastructures, firstly, the above challenge

(1) Can be addressed by providing local computing, storage, and networking resources to assist the often resource-poor IoT devices. The data generated by the edge devices at bewildering rates can be stored and preprocessed by the local edge cloud and only a small volume of processed data are required to be sent back to central datacenters. The networking load can be reduced. Secondly, for challenge

(2) The IoT devices can offload their tasks to the edge servers if the loads are beyond their capabilities. Since the edge cloud is closer to the devices, the latency can be well controlled compared to the conventional cloud computing model. Thirdly, for challenge

(3), an open edge cloud innovation platform can break the monopoly and accommodate fairer competition among all stakeholders, no matter if they are giant corporations or small or medium-sized inventors, vendors or ASPs. Specifically, these small or medium-sized stakeholders are usually closer to the common users and are the most active and innovative groups for Internet community. Such an open environment would help nurture future innovations.

To show how the conventional cloud computing and the new edge cloud computing differ in various aspects,

Table I. Brief comparisons between conventional cloud computing and edge cloud and edge computing.

Characteristics	Conventional cloud computing	Edge cloud and edge computing
Major applications	Most of the current mainstream cloud-involved applications	Applications on IoT, VR, AR, smart homes, smart cities, smart energy, smart vehicles, etc.
Availability	A small number of large-sized datacenters	A large number of small-sized datacenters
Proximity of services and resources; Data processing location	Usually in remote datacenters and far from users	At the edge close to the users
End-to-end latency	High, due to the distance between the edge and remote datacenters	Low, due to proximity to the users
Backbone network bandwidth consumption	High, since huge data need to be transferred to the datacenters first	Low, since data are locally processed and stored in edge cloud
Scalability	Scalable at center	Scalable both center and edge
Security (e.g., attacks on data enroute)	Data subject to attack due to long-distance transmission; Physical security depends on large facilities	Lower risk for enroute attacks; Physical security varies and different mechanisms needed

2. Edge vs. Fog computing

As the edge computing market takes shape, there's an important term related to edge that is catching on: fog computing.

Fog refers to the network connections between edge devices and the cloud. Edge, on the other hand, refers more specifically to the computational processes being done close to the edge devices. So, fog includes edge computing, but fog would also incorporate the network needed to get processed data to its final destination.

Backers of the *OpenFog* Consortium, an organization headed by Cisco, Intel, Microsoft, Dell EMC and academic institutions like Princeton and Purdue universities, are developing reference architectures for fog and edge computing deployments.

3. Edge computing security

There are two sides of the edge computing security coin. Some argue that security is theoretically better in an edge computing environment because data is not traveling over a network, and it's staying closer to where it was created. The less data in a corporate data center or cloud environment, the less data there is to be vulnerable if one of those environments is comprised.

The flip side of that is some believe edge computing is inherently less secure because the edge devices themselves can be more vulnerable. In designing any edge or fog computing deployment, therefore, security must be a paramount. Data encryption, access control and use of virtual private network tunneling are important elements in protecting edge computing systems.

4. Network Function Virtualization or NFV

Network functions virtualization (also Network function virtualization or NFV[1]) is a network architecture concept that uses the technologies of IT virtualization to virtualize entire classes of network node functions into building blocks that may connect, or chain together, to create communication services.

NFV relies upon, but differs from, traditional server-virtualization techniques, such as those used in enterprise IT. A virtualized network function, or VNF, may consist of one or more virtual machines running different software and processes, on top of standard high-volume servers, switches and storage devices, or even cloud computing infrastructure, instead of having custom hardware appliances for each network function.

Framework

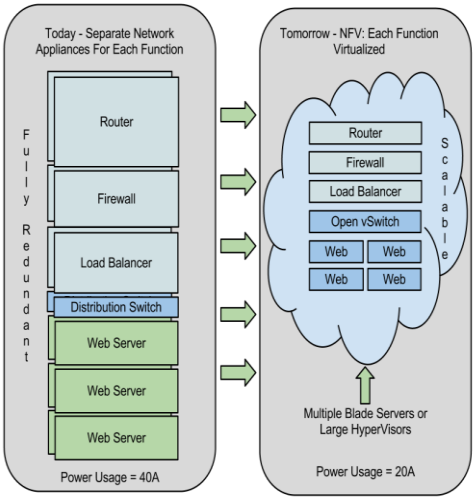
The NFV framework consists of three main components:[5]

1. Virtualized network functions (VNFs) are software implementations of network functions that can be deployed on a network functions virtualization infrastructure (NFVI).[6]
2. Network functions virtualization infrastructure (NFVI) is the totality of all hardware and software components that build the environment where VNFs are deployed. The NFV infrastructure can span several locations. The

network providing connectivity between these locations is considered as part of the NFV infrastructure.

3. Network functions virtualization management and orchestration architectural framework (NFV-MANO Architectural Framework) is the collection of all functional blocks, data repositories used by these blocks, and reference points and interfaces through which these functional blocks exchange information for the purpose of managing and orchestrating NFVI and VNFs.

The building block for both the NFVI and the NFV-MANO is the NFV platform. In the NFVI role, it consists of both virtual and physical processing and storage resources, and virtualization software. In its NFV-MANO role it consists of VNF and NFVI managers and virtualization software operating on a hardware controller. The NFV platform implements carrier-grade features used to manage and monitor the platform components, recover from failures and provide effective security – all required for the public carrier network.



Source: Steve Noble

Figure 2. Network Function Virtualization or NFV

5. Software-defined networking (SDN)

Software-defined networking (SDN) technology is a novel approach to cloud computing that facilitates network management and enables programmatically efficient network configuration in order to improve network performance and monitoring [1]. SDN is meant to address the fact that the static architecture of traditional networks is decentralized and complex while current networks require more flexibility and easy troubleshooting. SDN suggests to centralize network intelligence in one network component by disassociating the forwarding process of network packets (Data Plane) from the routing process (Control plane). The control plane consists of one or more controllers which are considered as the brain of SDN network where the whole intelligence is incorporated. However, the intelligence centralization has its own drawbacks when it comes to security [2], scalability and elasticity [3] and this is the main issue of SDN.

Software-defined networking (SDN) is an architecture purporting to be dynamic, manageable, cost-effective, and adaptable, seeking to be suitable for the high-bandwidth, dynamic nature of today's applications. SDN architectures decouple network control and forwarding functions, enabling network control to become directly programmable and the underlying infrastructure to be abstracted from applications and network services.[4]

The OpenFlow protocol can be used in SDN technologies. The SDN architecture is:

Directly programmable: Network control is directly programmable because it is decoupled from forwarding functions.

Agile: Abstracting control from forwarding lets administrators dynamically adjust network-wide traffic flow to meet changing needs.

Centrally managed: Network intelligence is (logically) centralized in software-based SDN controllers that maintain a global view of the

network, which appears to applications and policy engines as a single, logical switch.

- *Programmatically configured:* SDN lets network managers configure, manage, secure, and optimize network resources very quickly via dynamic, automated SDN programs, which they can write themselves because the programs do not depend on proprietary software.
- *Open standards-based and vendor-neutral:* When implemented through open standards, SDN simplifies network design and operation because instructions are provided by SDN controllers instead of multiple, vendor-specific devices and protocols.

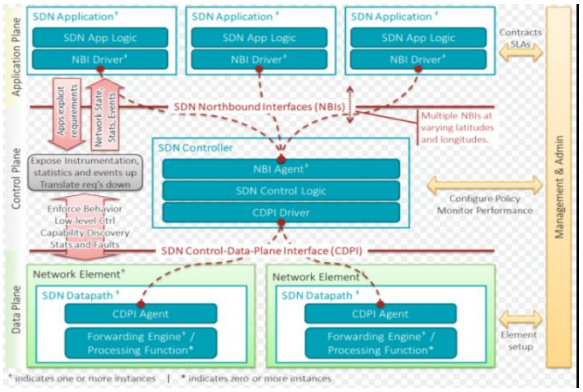


Figure 3. Software-defined networking (SDN)

SDN Application : SDN Applications are programs that explicitly, directly, and programmatically communicate their network requirements and desired network behavior to the SDN Controller via a northbound interface(NBI).

SDN Controller : The SDN Controller is a logically centralized entity in charge of (i) translating the requirements from the SDN Application layer down to the SDN Datapaths and (ii) providing the SDN Applications with an abstract view of the network (which may include statistics and events). An SDN Controller consists of one or more NBI Agents, the SDN Control Logic, and the Control to Data-Plane Interface (CDPI) driver. Definition as a logically centralized entity neither prescribes nor precludes implementation details such as the federation of multiple controllers, the hierarchical connection of controllers, communication interfaces between controllers, nor virtualization or slicing of network resources.

SDN Datapath: The SDN Datapath is a logical network device that exposes visibility and uncontested control over its advertised forwarding and data processing capabilities. An SDN Datapath comprises a CDPI agent and a set of one or more traffic forwarding engines and zero or more traffic processing functions.

SDN Control to Data-Plane Interface (CDPI): The SDN CDPI is the interface defined between an SDN Controller and an SDN Datapath, which provides at least (i) programmatic control of all forwarding operations, (ii) capabilities advertisement, (iii) statistics reporting, and (iv) event notification. One value of SDN lies in the expectation that the CDPI is implemented in an open, vendor-neutral and interoperable way.

SDN Northbound Interfaces (NBI): SDN NBIs are interfaces between SDN Applications and SDN Controllers and typically provide abstract network views and enable direct expression of network behavior and requirements. This may occur at any level of abstraction (latitude) and across different sets of functionality (longitude). One value of SDN lies in the expectation that these interfaces are implemented in an open, vendor-neutral and interoperable way.

6. Conclusion:

In this paper we presented various challenges with traditional cloud computing such as the traditional cloud infrastructure will run into a series of difficulties due to centralized computation, storage, and networking in a small number of datacenters, and due to the relative long distance between the edge devices and the remote datacenters To attack this challenge we proposed edge cloud and edge computing with emerging technologies including Network Function Virtualization (NFV) and Software Defined Networking (SDN).

References:

1. http://www.etsi.org/technologies-clusters/technologies/nfv

2. Benzekki Kamal et al.Devolving IEEE 802.1 X authentication capability to data plane in software- defined networking (SDN) architecture., Security and Communication Networks 9.17 (2016): 4369-4377.
3. Benzekki Kamal et al Software- defined networking (SDN): a survey., Security and Communication Networks 9, no. 18 (2016): 5803-5833.

4. "Software-Defined Networking (SDN) Definition". Opennetworking.org. Retrieved 26 October 2014.

5. Network-Functions Virtualization (NFV) Proofs of Concept; Framework, GS NFV-PER 002 v1.1.1 (2013-10),
6. Jump up^ "What is Network Function Virtualization (NFV)". blog.datapath.io.