International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-4, February 2020

A Comprehensive Research on Integration of Cloud with Software-Defined Network

Sanam Fayaz Sahito, Samreen Detho, Syed Asif Raza Shah, Tae-Hyung Kim, Seo-Young Noh

Abstract: Cloud computing, as one of the emerging technologies, is required to meet increasing and on-demand availability of all resources such as computing power, storage and networking. It provides the pool of resources and offers the mechanism to run various applications on Cloud Data Centers (CDCs). Cloud data centers are usually distributed across multiple sites. With the provision of distributed data centers, the need for their management and maintenance arises. Such circumstances lead networking capabilities to being implemented on larger scale. Software defined network (SDN) is the programming paradigm along with the NFV orchestration and incorporates the modular and dynamic network support for the cloud data centers provided and established by the cloud over the geographically separated places. In this paper, comprehensive study has been mainly conducted to highlight the need for the integration of both emerging and versatile technologies. We also cover challenges, issues and benefits which have to be considered in underlying architecture, models and devices.

Keywords: Software Defined Network (SDN), Network Function Virtualization (NFV), NFV Orchestration, Cloud Data Centers (CDCs)

I. INTRODUCTION

Cloud provides the variety of services to its customers on the "Pay as you go" principle [1]. With the rapid growth of data and high demand to such services, Cloud has grasped the large amount of customers. Though the community of cloud users is increasing day by day, provision of the cloud services must be responsive at high rate and adaptive against each changed requirement [2]. Cloud providers mainly establish the infrastructure for the storage and hardware related services based on the data centers. Cloud data centers are developed and maintained remotely over the distributed areas to store the resources and provide the hardware support to the users.

Revised Manuscript Received on February 28, 2020.
* Correspondence Author
Sanam Fayaz Sahito, Department of Computer Science, Sukkur IBA University, Sindh, Pakistan. Email: sanam.khp@iba-suk.edu.pk
Samreen Detho, Department of Computer Science, Sukkur IBA University, Sindh, Pakistan. Email: samreen.mscs17@iba-suk.edu.pk
Syed Asif Raza Shah, Department of Computer Science, Sukkur IBA University, Sindh, Pakistan. Email: asif.shah@iba-suk.edu.pk
Tae-Hyung Kim, Samsung Electronics, Seoul, Korea. Email: thkim4u@hanmail.net
Seo-Young Noh*, Department of Computer Science, Chungbuk National University, Cheongju, Korea. Email: rsyoung@cbnu.ac.kr

There are many software tools to manage cloud-based infrastructure. One of popular software tools is OpenStack.

OpenStack [3] is the open source platform that usually manage and build the clouds over the network. It can control entire set of resources and services of clouds within the underlying data centers. It basically enables the orchestration of the all cloud based resources and services. By handling the multiple hypervisors, OpenStack has capability to manage various cloud based data centers at the same time. As the scale of a data center is growing, it becomes more time-consuming and technically challenging work for the cloud providers to manage and maintain entire infrastructure with OpenStack.

Software defined networking is the full fledge programming paradigm that enables any network to be scalable dynamically with incorporation of programming techniques. Integration of all these robust technologies such as Cloud, OpenStack and Software defined network results in a versatile paradigm that is dynamic, adaptive, scalable, and reliable at the same time. In order to achieve better consequences, the proper virtualization support must be incorporated. Network function virtualization is a technology used to virtualize network services such as: load balancing, routing, and firewalls. NFV is capable of adding new network functions and applications. It provides the virtualization support to the networks by facilitating modular virtualization of the all required network devices. In this regards, different opportunities and challenges can be encountered. Firstly SDN facilitates the decoupling for its two separated planes data pane and control plane. Dynamically scaling up is another benefit offered by the SDN. OpenStack make it possible to control the orchestration of the set of resources.

Section 2 discusses cloud computing technology in detail. Wherein detailed information of cloud deployment models, three main services of cloud and limitation of cloud data centers are discussed. Section 3 covers Software defined networks, its architecture and essential components to work with SDN. In continuation with SDN discussion, section 4 provides brief overview of one of the open source cloud platform known as OpenStack. Section 5 elaborates integration of SDN with cloud. This section also covers benefits of cloud along with common issues and challenges which are currently faced by cloud users. In section 6, we will discuss available platforms for simulation of SDN with cloud. Furthermore, comparative analysis of mentioned simulators/emulators is presented. Section 7 covers features and challenges of SDN enabled cloud. Section 8 concludes whole article and addresses the contribution of this research.

II. BACKGROUND

A. Cloud Computing

With the utilization of the Internet, the cloud computing facilitates the accessibility of resources on large scale, as shown in Figure 1. For instance, resources can be of any type such as computing, network connectivity or storage resources. Cloud makes sure the availability of the resources on the "pay as per use" scheme. Customers of cloud can access their required resources by on-demand through the network [4].

Cloud works on the principle of "Anything, Anytime and Anywhere." To establish the cloud infrastructure, various services and deployment models have been built- up.

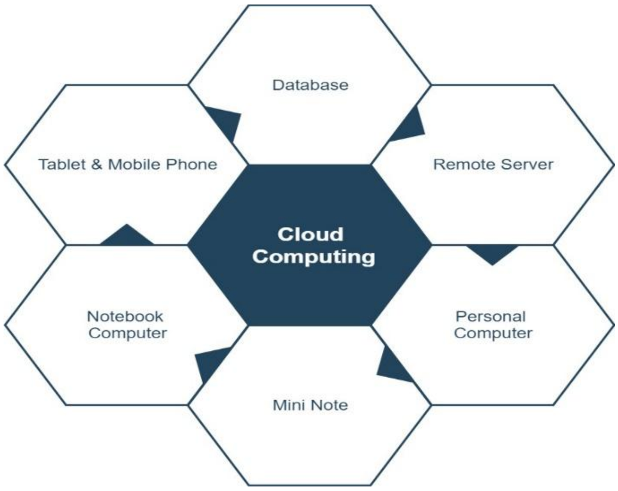


Figure 1: Cloud Computing

National Institute of Standards and Technology (NIST) [5] introduced four main deployment models of cloud. Each deployment model differ based on an infrastructure the cloud uses and its place where it is located.

As shown in Figure 2, there are mainly four deployment models that usually cloud offers and builds for their customers, and on the basis of these models cloud makes available all the services to its community of customers.

- ***Private Cloud:*** This model refers to an infrastructure that is mainly dedicated to a specific entity, an enterprise or organization. The major advantage that this type of cloud offers is the security because of the deployment of cloud at the place (premises) of its customer.
- ***Public Cloud:*** This deployment model offers numerous types of resources to its users in the form of cloud services that can be accessible and available remotely. Benefit that can be achieved with these models is that cloud users do not have to invest initial cost to set up the infrastructure.
- ***Hybrid Cloud:*** Hybrid model is the merger of two model discussed above and offers the proper orchestration of resources utilized between private and public models of deployment. "Hybrid deployment models" usually are more flexible than the others.
- ***Community Cloud:*** Several organization with the similar and specific goal build and utilize the single infrastructure. Such an infrastructure shared among those various organizations is called "community". The main purpose of this model is to build up a cloud service accessed across multiple organizations located in geographically or politically different boundaries.

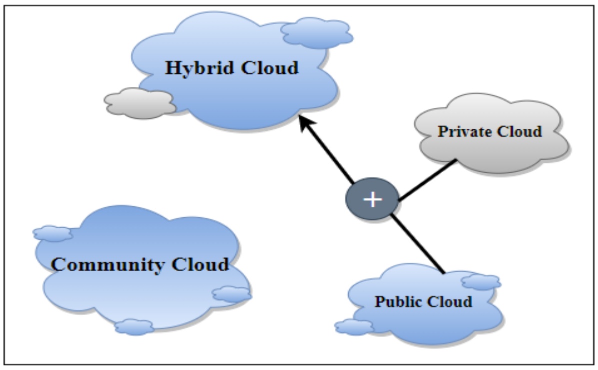


Figure 2: Deployment Models of Cloud Computing

As shown in Figure 3, cloud service providers ensures the provision of the highly "responsive", "adaptive" and "available" services based on the customer requirements [4]. Customers usually have to pay only up to the consumption of the resources.

- ***Software-as-a-Service (SaaS):*** Cloud offered service model, in which the software and applications are usually built and provided by the third party hosts. Then, those applications are being made accessible and available to the cloud customers on the network referred as the Internet. They are licensed and hosted centrally. Most of the SaaS applications are accessible via thin clients. These services are also known as "on-demand software". One of the most commonly and widely used Software-as-a-Service is Google Apps. Users can access the services like drive, Gmail, Docs and photos.
- ***Platform as a Service (PaaS)***: Platform is being offered in these types of services to the cloud customers. These platforms usually facilitate the implementation of their business applications, and then its management and execution is also possible. Mainly these services fulfill the need of platform provisioning where one can develop, run and deploy their applications. Due to infrastructure's availability, programmers and users do not need to worry about building and maintaining infrastructure. Typically building and launching of applications are major usage of Platform-as-a-Service. Most of platforms include Operating System, database, programming language environment for execution and multiple web servers. AWS Elastic Beanstalk, Microsoft Azure and Google App Engine are common examples of PaaS.
- ***Infrastructure as a Service (IaaS)***: Outsourcing the resource like hardware as well as storage are known as the infrastructure as a service model. IaaS provides high-level application programming interfaces used to link details of given underlying network, location provisioning, security and backup infrastructure. IaaS considered as whole new setup for a business without collection of physical devices. Cisco Metacloud, AWS, Google Compute Engine and Rackspace cloud are known as Infrastructure-as-a-Service.

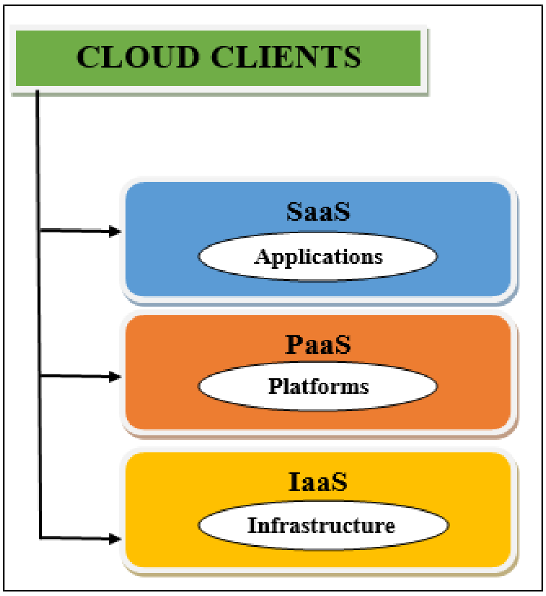


Figure 3: Services Offered by Cloud Provider

In the continuation with discussed services, even though cloud may offer many different types of services, these three type explained above are the most demanded services until now.

Cloud utilizes the mechanism of the cloud based data centers in order to make available anytime anywhere and keep highly responsive to its services over the network. As with the rapid growing of community of cloud customers the numerous data centers have been established into the cloud network infrastructure. The data centers allow applications to become more adaptive with the rapidly changing requirements. In this regard, OpenStack provides controlled orchestration of the cloud. Its established infrastructure is based on the network of data centers, but it possesses some limitation too. OpenStack also has specified range but it is not so much scalable in order to keep up with today's changing requirements. There is a need to develop dynamic network that possesses the ability to grow timely and rapidly. Therefore the integration of software defined network with the cloud network can enhance its capabilities to pace with the current requirements.

B. Software-Defined Network (SDN)

SDN is a versatile technology that offers numerous benefits. For instance, the networks can be decoupled and become dynamic and flexible [6]. Its programmatic architecture facilitates the scalability of the networks and automated resource orchestration. Two planes (or stacks) mainly work for a network to be built, one is "data plane" and the other is "control plane". Before SDN, both of the interfaces (planes) were tightly coupled. However, SDN has introduced the concept of decoupling these two phases, leading to the way of the provision of the "decoupled control". Furthermore SDN uses the "Open Flow Protocol" to enable the communication between both of these planes.

Control plane is the one of SDN's stacks that contains a controller that usually makes all the decisions related to the transmission medium, routing, and orchestration. This plane has all those functions as well as processes which are responsible for selecting packet forwarding route and to make all decisions related to the controlling, management, and traffic. It also enables the signals of the networks. Controller is basically a logical component of SDN architecture, it gets directions from the upper layer. Information is usually extracted from the network components with the help of controller, and the interface that holds the network components resides below the control plane.

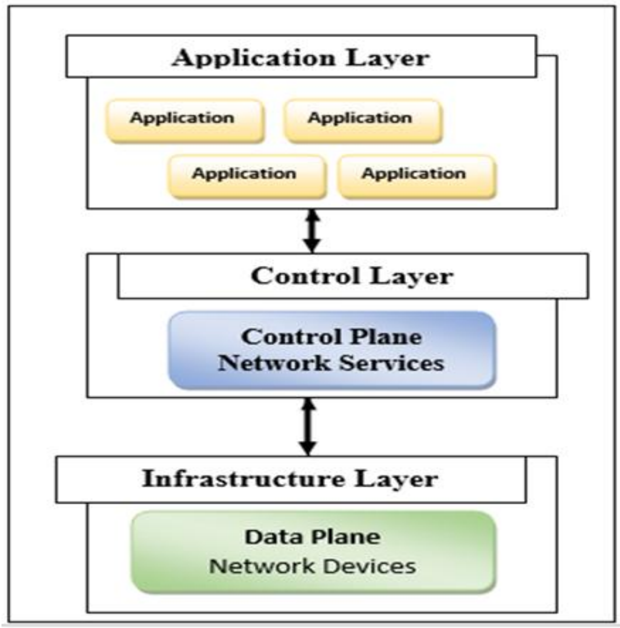


Figure 4: SDN Architecture

Data plane, sometime called forwarding plane or user plane is responsible for forwarding the packets or data up to the immediate hop. Data packets usually get forwarded through the device termed as the router.

- ***SDN Architecture:*** As shown in Figure 4, Software-defined network's architecture contains three main stacks referred as the components such as, "SDN controller", "Networking devices" and "Application layer." The application layer usually contains the programming modules that are linked with the controller to produce the signals for the resources in need. SDN controller is the logical layer. All tasks related to the signaling and controlling are being performed in this layer. Controller resides here to get instructions directly from the upper layer which is the application layer. SDN networking components layer represents all the physical components required for the networking. Data plane resides in this layer. Therefore, this layer is responsible for packet forwarding as well as for enabling the processing. The southbound and northbound APIs are used to establish connections between all three architectural layers of the SDN. Communication between applications and controller is enabled by the interface termed as the northbound API and the other one. The southbound API establishes the connection between the controller and its lower layer referred as the network components layer.

- ***OpenFlow Switch Architecture:*** SDN adopts OpenFlow protocol to communicate between the infrastructure layer and the control layer. It works like a communicating bridge between a network switch and a controller. OpenFlow has some additional functionalities such as power saving, load balancing, monitoring and mechanism [7]. Given architecture and result show that SDN-based environment can provide Quality of Service (QoS) that should be guaranteed for cloud computing services. An OpenFlow switch can be hardware or software where OpenFlow protocol is implemented. In order to build agile and flexible network by using SDN, all the computing devices with OpenFlow switch are connected. The OpenFlow switch separates data path and controller. The architecture is composed of controller, secure channel, group table and flow tables. Figure 5 illustrates architecture of OpenFlow Switch. This upper layer is known as Controller. Lower layer contains OpenvSwitch. Overall system can have multiple virtual clusters composed of number of virtual machines that communicate via OpenvSwitch.

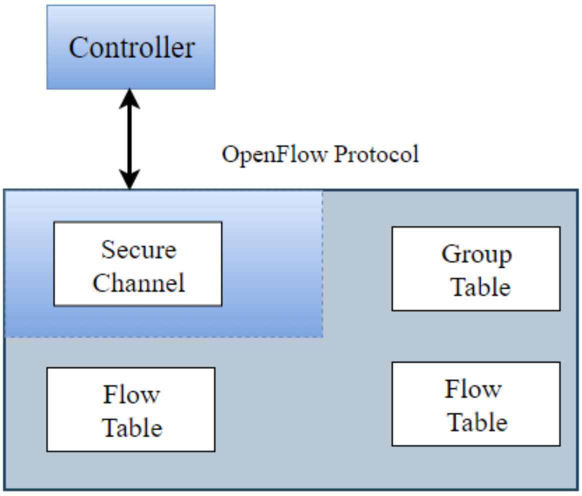


Figure 5: The architecture of OpenFlow Switch

- **Benefits and Challenges:** Moving towards the software defined networking, while configuring the networks, many favors can be gained, but a few challenges can be encountered. Firstly, it controls the overall network by programming. Decoupling the control plane from the data plane is the major benefit offered by the SDN. Secondly, it enables the virtualization and provides the automated orchestration at the same time for the network components with the incorporation of NFV. Thirdly it enhances the dynamic scalability and optimization of the network resources. Interoperability, controller placement and security are key challenges for moving towards the SDN platform. Complete shifting can be proved more painful due to less knowledge of implementation and management. To achieve performance enhancement can be another challenge, while proper support of APIs is not fully standardized yet. Implementing controller can be more hectic task to be done.

III. INTEGRATION OF SDN WITH CLOUD COMPUTING

Cloud computing is considered as paradigm shift of traditional networks towards ubiquitous services. It enhances the way of accessing and storing data. Initially cloud were used only for storage purpose. Cloud was considered as the platform for managing, storing and accessing data over the internet. As technology evolves, cloud get major boost by providing compute services, network services, storage and automation services as well. Now, users can get a complete application, platform and infrastructure over the internet. Cloud has multiple servers that are accesses remotely in order to make its services available. Some common issues of cloud such as performance, management, scalability, and network reachability can be solved with the integration of Software Defined Network (SDN). Control plane and data plane have been separated in SDN. They manage IP assignment, network allocation, service testing and network reachability (end to end reachability) in cloud networks. Whole routing changes and bandwidth utilization is also tackled by SDN. As the orchestration is performed in some of cloud platforms so it makes the infrastructures more reliable, scalable and efficient. SDN manages network equipment flexibly by running software on external servers [8]. Decoupling data plane and control plane accomplishes the cloud computing vision of utility computing [8]. Cloud computing with SDN eliminates up-front cost for networking infrastructure, reduces maintenance cost and provides on-demand computational and networking scaling services. Network virtualization and orchestration in SDN enable cloud to work more efficiently [8]. In particular, OpenFlow [9] provides a standard interface for caching packet forwarding rules in the flow table of switches, querying traffic statistics, and notifying topology changes. OpenFlow allows infrastructure layer of SDN to communicate with Control layer [9]. Direct access has been allowed by OpenFlow for manipulation of forwarding plane of network devices such as switches and routers, both physical and virtual (hypervisor-based).

SDN makes network programmable and devices can work with new standards. Cloud computing and SDN complement each other provide control of network.

A. Benefits of SDN for Cloud

SDN is considered as the solution for cloud service providers. While expanding the network, there is no need to worry about additional control planes. Minimum planning is required at times of expanding a network. Without SDN, reconfiguration of each device will be required to meet the user needs. Several networks can be managed by cloud providers without splitting them into multiple VLANs as SDN virtualizes whole environment. There is central management with distributed environment, which provides network administrator with the ease of management. Following are major benefits of SDN for Cloud Computing:

- ***Cost Reduction:*** SDN supports networking functionalities of layer 1, layer 2 and layer 3 devices through software networks. No additional cost is required for this setup as many of SDN services are free. License should be paid for some SDN solutions such as VMware's NSX. Multiple operating environments such as OpenDaylight, ONOS and Floodlight work with SDN.
- ***Intelligent Global Connection:*** SDN works with load-balancing cloud and infrastructures of data centers for creating intelligent and globally connected environments. SDN uses network logic for global traffic management and sending traffic to authorized data centers. Integrating SDN with cloud platforms enables architects to create fluid automation of traffic flows in data centers.
- ***Reduced Downtime:*** Upgradation of one device that is virtualized in multiple nodes consumes less downtime than upgrading several devices. It also helps to recover from any fault that has been caused by upgrades.
- ***Granular Security:*** In virtualized environment, network management becomes difficult to do. Applying firewalls and filtering policies on network becomes difficult as well. SDN provides central point of control to distribute security information within network organization.

B. Issues and Challenges

Integrating SDN in cloud platform is very beneficial but many challenges and issues occurred during this process. Those challenges hinder the performance and implementation. A few of those challenges are discussed below:

- ***Controller Placement:*** Deciding the location of controller in SDN is threefold problem. It depends upon number of controllers. A controller related with multiple switches and placement requires efficient network design plan [10].
- ***Security Issues:*** In public clouds, implementing SDN can come up with security issues. Some attacks like Denial of Service, Spoofing and malicious injections are very common in virtualized networks [11]. Control plane decoupled with data forwarding hardware makes it vulnerable to attacks.
- ***Bugs in Large Scale Distributed System:*** Cloud is composed of distributed nature of Virtual Machines that enlarge the network and SDN establish centralized controllers for managing such huge frameworks. If any of the bug encompasses, it is difficult to find out easily so there is need of debugger that relies on multiple VMs [12].
- **Performance:** There is explosive growth in applications and user requirement of cloud computing. Performance can be deprived by scaling the network, data transfer and data lock-in. It is usually measured in terms of service and application availability.
- ***Lack of Resources and Expertise:*** SDN with cloud is the latest technology emerging day by day. It has complete paradigm shift from traditional networks. Implementing such a technology requires special expertise of networks, virtualization, IP allocation and reachability. Generic hardware resources are required for this integration. Virtual networks are easy to manage over cloud once created but virtual machines implementation requires additional computer-related skills.

There are several other issues of integrating SDN with cloud computing i.e. scalability, application availability, interoperability and reliability. The whole network depends upon central controller, if any issue occurs then whole network collapses. In order to decrease the chances of controller failure, cloud must decide the functionality and utilization of controller [13]. If there is any link failure then controller must support multi-path solutions and traffic rerouting. There is need of standard APIs so that data plane and control plane will communicate with each other.

C. Features

Most useful and responsive network infrastructure can be built with Software Defined Networks. It removes the implementation and scaling of new hardware for additional tasks.

Deploying SDN with cloud provides agility for IT manager with cost-effective implementation. It allows to address the security issues in a better way. In a traditional network architecture, organizations have to buy new devices every year but with SDN most of the hardware based services are converted into software applications. SDN has capability of auto configuration. It can reconfigure on the fly as per need of application. This integration could be considered as disruptive evolution of business towards expense-centric models in typical clouds. SDN enabled clouds bring portability and programmability in network infrastructures [14]. Users can work on orchestration of applications using SDN in cloud. Main features of Integrating SDN with cloud are listed below:

- Cost Efficiency
- Application Optimization
- Orchestration
- Scalability
- Virtualization
- Centralized Controlled
- Decoupled Data and Control Plane
- Network Policy
- Network Reachability
- Dynamic Network

IV. SDN ENABLED OPENSTACK CLOUD

Neutron is the replacement of the networking API termed as the "Quantum". It is referred as the networking controller for the OpenStack [11]. Neutron facilitates the development of the SDN over the network based on cloud infrastructure. Neutron project is initiated by the OpenStack, it enables the utilization of the "Network as a Service". To get integrated both of the technologies, the neutron agent and the controller plug-in must be linked together. Figure 6 shows the overall architecture for the underlying integration model. Both technologies must be interfaced with each other to enable communication.

Neutron has its logical architecture [15], while working with OpenStack it receives the instructions from all other components of the OpenStack. Then, "neutron server" establishes the virtualized network and initiates signals to the underlying plug-ins to handle received requests.

Neutron Plugins are responsible for establishing the logical link between the neutron server and database, and forwarding the request towards the neutron agent. Such an agent ultimately executes the request on the "network provider" [15]. This Network Provider is basically responsible for the service implementation such as OpenvSwitch (OVS) or any sort of networking switch. Message queue is mainly responsible for controlling the overall communication.

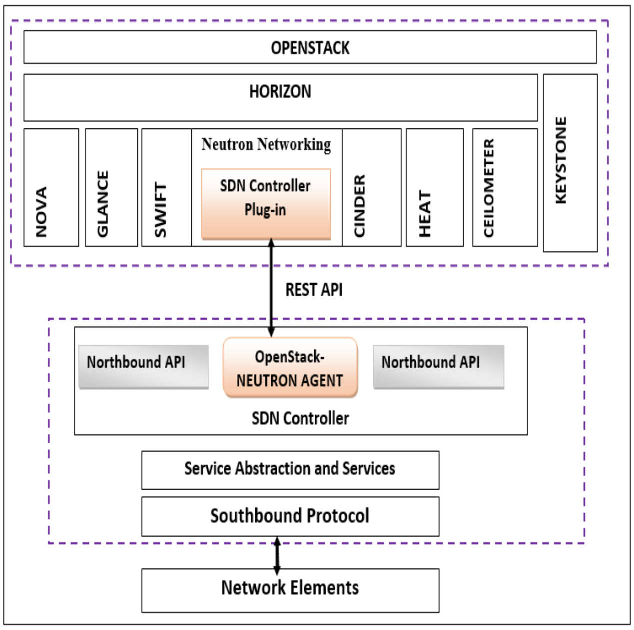


Figure 6: SDN Enabled OpenStack Cloud

Integration process of OpenStack with the SDN lies into two stages. First one is to enable the virtualization. For this, SONA (Simplified Overlay Network Architecture) can be used to facilitate virtualized network. Second one is to implement the data plane for the networking components linked with the physical layer of the architecture. Finally the ONOS (Open Network Operating System) controller would be incorporated to control and manage the both physical and logical planes.

OpenStack is a cloud platform that manages huge compute, storage and network resources. All resources are managed through APIs with common authentication mechanism. It provides the features of automation and orchestration to manage multiple applications over cloud. It is well defined cloud platform. It is usually deployed as Infrastructure-as-a-Service. SDN is composed of three sections: Network management, Control Plane, and Data Plane.

OpenStack cloud has 9 main components in the implementation of SDN. A few challenges have to be addressed. A network management through horizon dashboards needs specialized skills. Separating data plane and control plane requires multiple VMs to process data for flow and storage. Special database known as Trove is used to store data in the form of relational and non-relational structure. Trove is reliable and scalable database service of cloud [13].

Neutron provides networking service for OpenStack so that control plane of SDN must work with it. Neutron ensures that each component of the network must communicate with each other properly. Glance retrieves and stores virtual machine disk images. Nova is OpenStack service that provide computational capability in order to fulfill requirements of core computational operations.

V. SIMULATION PLATFORMS

Cloud computing offers great advantages and services. Grid computing, utility computing, web computing and virtualization, all are part of cloud these days. Most of these features enable users to work in cloud environment. It is the most powerful tool to work on High Performance Computing (HPC). End user can be provided with applications and services from cloud provider over the internet. Cloud providers need to evaluate their services time to time. If those evaluations need to be in real-time, then the whole process becomes so costly. In order to reduce this issue of cost, simulation platforms provide ease of use. According to literature there are number of network modeling and simulation platforms which are based on the concept of cloud computing. "Cloud Simulation" platform can improve the capability of simulation of grid in sharing, collaborating, fault-tolerating and migrating multi-granularity resources on demand by multi-user, which establishes a new modeling and simulation mode [16]. There are multiple other platforms for simulation of cloud such as "CloudSim": It is a framework for modeling and simulation of cloud services and infrastructure. Multiple platforms for Software Defined Network simulation are also available.

A. Open Source Cloud Computing Simulators

For implementing new algorithms, standards, testing and experiments, cloud based real environments are costly and difficult to create. But it is necessary to test before implementation. Cloud Computing Simulators play a vital role for reducing development cost, incorporate with advancements within technology, analyze security threats, and reduce the complexity and overall performance of the infrastructure. Simulators provide ease-of-use, facilitate flexible and dynamic configuration and development environments. Following are some Open Source and free cloud simulators that perform better than some commercialized ones.

- **CloudSim –** It is a Java-based simulation toolkit that is generalized and extensively used by researcher and cloud community. It supports communication between components, and creates CloudSim entities, queuing events and manage the simulation clock. Features of CloudSim include virtual machines, applications, users, data centers and computational resources. It supports simulation for large scale cloud data centers.

- **CloudAnalyst –** This simulator is derived from CloudSim with some additional features. CloudAnalyst supports the evaluation of social networking tools according to the geographical distribution of users and data centers. Determining the behavior of large scale internet applications is possible with this tool. It can conduct multiple simulations with little change in parameters. Monitoring cloud clusters, data flow in centers and load balancing as real-time environment is supported by CloudAnalyst. Features of CloudAnalyst includes graphical User Interface for output and setup, efficient output and experiment looping.
- **iCanCloud –** It supports the simulation for the networks having large storage. This simulation framework is based on SIMCAN that predicts the tradeoff between cost and performance of set of applications that are executed in specific hardware. It provides detailed information about cost to user. iCanCloud was designed specifically for optimizing flexibility, performance, accuracy and scalability. It can model both existing and non-existing cloud computing architectures. Customizable Virtual Machines can used to simulate uni-core and multi-core systems quickly.
- **GreenCloud –** This platform has been developed for energy-aware data centers of cloud computing. It is considered as the best packet level simulators that enable communication in clouds. This simulator was developed for workload scheduling, monitoring and resource allocation. GreenCloud is considered as the solution for communication protocol, optimization and network infrastructure. It provides support for simulation of CPU, storage, memory and network resources is available in mentioned platform.
- **EMUSIM –** It is the combination of AEF (Automated Emulation Framework) and CloudSim. It extracts information of application behavior automatically. The EMUSIM simulator is of great use when the tester has no idea about the performance of the software under the various levels of concurrency and parallelism which impede simulation. Main features of EMUSIM includes providing the information of performance, combining emulator and simulator, and supporting loosely coupled CPU intensive applications.
- **GroundSim –** This Java-based simulator is designed for scientific applications that need event-based results. There is a need of only one simulation thread. It is a simulation tool for Grid and cloud computing as well. It can easily simulate complex scenarios. It mainly focuses on IaaS that can be extended to PaaS and SaaS.
- **DCSim –** This extensible data center simulator is designed using Java. High-end experiments on data center management techniques and algorithms can be developed using DCSim. It contains a multi-tier application model that allows the simulation of dependencies between VMs [17].

Table 1 shows the comparison among these simulators on the basis of multiple features discussed in [17].

Table 1: Comparison of Open Source Cloud Computing Simulator

Simulator	Platform	Type	Programming Language	Cost Modelling	GUI	Communica tion Model	Simulation Time
CloudSim	SimJava	Open source	Java	Yes	No	Limited	Second
CloudAnalyst	CloudSim	Open source	Java	Yes	No	Limited	Second
GreenCloud	NS-2	Open source	C++, oTCL	No	Limited	Full	Minute
iCanCloud	SIMCAN	Open source	C++	Yes	Yes	Full	Second
EMUSIM	CloudSim ,AEF	Open source	Java	Yes	No	Limited	Second
GroundSim	_	Open source	Java	No	Limited	No	Second
DCSim	_	Open source	Java	No	Limited	No	Second

B. SDN Emulators and Simulators

There are 2 main emulators available for software defined networking.

- **Mininet Emulator –** It creates a network of virtual hosts, controller, switches and links. Switches in mininet support OpenFlow standard protocol for communication. OpenFlow supports highly flexible routing and Software Defined Networking. This emulator is used for deploying large networks on limited resource machine having VMs. Mininet emulator can execute unmodified code interactively on virtual hardware on a simple PC [18]. It provides realism and convenience at low cost. It allows users to create topologies with thousands of nodes and multiple hosts. User is also able to work in a way that data plane and control plane still remain separated. In general it runs on Linux environment. It allows users to build, customize, share and test their own SDN. It emulates OpenFlow devices and controllers. It contains many default topologies such as single, minimal, reversed and tree [18].
- **EstiNet Simulator –** It uses the kernel reentering simulation methodology that enables real applications to run on nodes in Simulink network. It works as both emulator and simulator. It combines the advantages of emulation and simulation approaches without their respective shortcomings [19]. This tool is also used for working on Software Defined Networks, supporting OpenFlow to work on SDN. In the article [20], the author compares Mininet and EstiNet on the basis of correctness, performance and scalability. Floodlight OpenFlow controller is used without any modification. Experiments have been conducted on grid network and the EstiNet correctness results are always correct and repeatable. Mininet requires time consuming effort for configuring large networks as compared to EstiNet. It is suggested that scalable and large networks are better to simulate in EstiNet.
- **NS-3 Simulator –** It is a discrete event-network simulator. It is used for academia and research and helps user with their work on SDN through networking simulator. It works accurately at times of testing Software Defined Networks. Spanning trees in SDN can be built using NS-3. SDN enables flexibility in communication networks as there is separation of globally defined standards [21]. NS-3 can work fine with programmable application of SDN.

VI. CONCLUSION

In this paper, two technologies i.e. Cloud Computing and Software Defined Networks are discussed in detail. From emergence to the integration, each component of both technologies are discussed. The aim of this paper is to aware community about latest trends, platforms and technologies that are using Cloud Computing with SDN. Cloud is the best platform for initializing any IT business with low costs. SDN enables programmable environment to work with. Open Source platforms and simulators for implementing cloud computing with SDN are also presented.

In this paper, a comprehensive study has been mainly conducted to highlight the need for the integration of both emerging and versatile technologies. In addition, major challenges, issues and benefits are presented with consideration of underlying architectures, models and devices used in SDN.

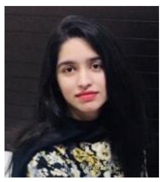
ACKNOWLEDGMENT

This work was supported by the research grant of the Chungbuk National University in 2019.

REFERENCES

1. Banikazemi, Mohammad, et al. "Meridian: an SDN platform for cloud network services." IEEE Communications Magazine 51.2 (2013): 120-127.
2. Shieh, Alan, et al. "Seawall: Performance Isolation for Cloud Datacenter Networks." HotCloud. 2010.
3. Mell, Peter, and Tim Grance. "The NIST definition of cloud computing." (2011).
4. Hazarika, Bhaswati, and Thoudam Johnson Singh. "Survey paper on cloud computing & cloud monitoring: basics." SSRG Int. J. Comput. Sci. Eng 2.1 (2015): 10-15.
5. Kumar, Rakesh, et al. "Open source solution for cloud computing platform using OpenStack." International Journal of Computer Science and Mobile Computing 3.5 (2014): 89-98.
6. Arlimatti, Shivaleela, et al. "SOFTWARE DEFINED NETWORK AND OPENFLOW: A CRITICAL." (2006).
7. Yen, Tseng-Chang, and Chi-Sheng Su. "An SDN-based cloud computing architecture and its mathematical model." 2014 International Conference on Information Science, Electronics and Electrical Engineering. Vol. 3. IEEE, 2014.
8. Azodolmolky, Siamak & Wieder, Philipp & Yahyapour, Ramin, "SDN-based Cloud Computing Networking (Invited)", ICTON 2013.
9. McKeown, Nick, et al. "OpenFlow: enabling innovation in campus networks." ACM SIGCOMM Computer Communication Review 38.2 (2008): 69-74.
10. Dvir, Amit, Yoram Haddad, and Aviram Zilberman. "The controller placement problem for wireless SDN." Wireless Networks 25.8 (2019): 4963-4978.
11. Mabel, J. Prathima, K. A. Vani, and KN Rama Mohan Babu. "SDN Security: Challenges and Solutions." Emerging Research in Electronics, Computer Science and Technology. Springer, Singapore, 2019. 837-848.
12. Paliwal, Shailesh. "Performance challenges in cloud computing." (2014).
13. Shamugam, Veeramani, et al. "Software Defined Networking challenges and future direction: A case study of implementing SDN features on OpenStack private cloud." IOP Conference Series: Materials Science and Engineering. Vol. 121. No. 1. IOP Publishing, 2016.
14. David Linthicum, "Pair SDN with cloud for more efficient network management".
15. He, Mu, et al. "Evaluating the Control and Management Traffic in OpenStack Cloud with SDN." IEEE International Conference on High Performance Switching and Routing (HPSR). 2019.
16. Li, Bo Hu, et al. "Networked modeling & simulation platform based on concept of cloud computing-cloud simulation platform." Journal of System Simulation 21.17 (2009): 5292-5299.
17. Nayyar, A. "The Best Open Source Cloud Computing Simulators."OpenSource site (2016).
18. Kaur, Karamjeet, Japinder Singh, and Navtej Singh Ghumman. "Mininet as software defined networking testing platform." International Conference on Communication, Computing & Systems (ICCCS). 2014.
19. Wang, Shie-Yuan, Chih-Liang Chou, and Chun-Ming Yang. "EstiNet openflow network simulator and emulator." IEEE Communications Magazine 51.9 (2013): 110-117.
20. Wang, Shie-Yuan. "Comparison of SDN OpenFlow network simulator and emulators: EstiNet vs. Mininet." 2014 IEEE Symposium on Computers and Communications (ISCC). IEEE, 2014.
21. Jared S. Ivey, Michael K. Riley, and George F. Riley, "A Software-Defined Spanning Tree Application for ns-3".
22. Nunes, B.; Mendonca, M.; Nguyen, X.; Obraczka, K.; Turletti, T., "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks," Communications Surveys & Tutorials, IEEE, Vol. 1, No.99, pp. 18.
23. Furht B and Escalante A 2010 Handbook of cloud computing (New York: Springer)
24. Jammal M, Singh T, Shami A, Asal R and Li Y 2014 Software defined networking: State of the art and research challenges Computer Networks vol 72 p 74-98.
25. Sefraoui, Omar, Mohammed Aissaoui, and Mohsine Eleuldj. "OpenStack: toward an open-source solution for cloud computing." International Journal of Computer Applications 55.3 (2012): 38-42.
26. De Oliveira, Rogério Leão Santos, et al. "Using mininet for emulation and prototyping software-defined networks." 2014 IEEE Colombian Conference on Communications and Computing (COLCOM). Ieee, 2014.
27. Mijumbi, Rashid, et al. "Network function virtualization: State-of-the-art and research challenges." IEEE Communications surveys & tutorials 18.1 (2015): 236-262.
28. Li, Yong, and Min Chen. "Software-defined network function virtualization: A survey." IEEE Access 3 (2015): 2542-2553.

AUTHORS PROFILE



Sanam Fayaz Sahito received the B.S. degree in Computer Science from Shah Abdul Latif University Khairpur in the year 2017 with the distinction of 3.94 CGPA out of 4.0, and the M.S. degree in Net Centric Computing (NCC) from Sukkur IBA University in 2019. She is a network specialist with major interest in Software Defined Networking, Cyber Security, and Internet of Things. Her research publications are multidisciplinary. She has interest in developing and enhancing pedagogical strategies of integrating ICT with other fields. She served 1.5 year as ICT teacher at The City School Indus Campus Sukkur. She was master trainer and conducted number of trainings on "Implementation and Integration of ICT". Currently, she is working as the faculty member of Sukkur IBA University. Her research interests include Computer Networks, SDN, Software Engineering, Neural Networks, Internet of Things, and Cyber Security.

Samreen Detho received her Bachelor of Science degree in Computer Science with the Specialization in Software Engineering from Sukkur IBA University with 3.14 CGPA and recently completed her Master of Science in Computer Science from Sukkur IBA in 2019, having major Interests in Data Science and Computer Networks specifically Big Data, Machine Learning, Multimedia & Image Mining, Cloud Computing and Internet of Things. Currently she is serving for the Profession of teaching as the Lecturer of Computer Science at IBA Community college Jacobabad since 2017. She has mainly worked on the projects related to the data science for instance Image, Video & Audio Processing, Prediction tasks using Deep Learning.

Dr. Syed Asif Raza Shah is working as Assistant Professor of Computer Science department at Sukkur IBA University, Pakistan. He has earned his PhD in Computer Science in 2019, from University of Science and Technology (UST), South Korea. He has been involved in different scientific data management research projects at Korea Institute of Science and Technology Information (KISTI) during his PhD research. He also has more than 10 years of industry experience in the fields of networking and system. Based on his research publication track record, he has contributed in the areas of Software-Defined Networks (SDN), Cloud Computing, Virtualization, Network Function Virtualization (NFV), Blockchain, Security, and Scientific distributed computing.

Dr. Tae-Hyung Kim received the master's degree in computer science from New York University and the PhD degree in computer science from Iowa State University. He is currently a principal software engineer at the Samsung Research, Samsung Electronics. His research interests include software architecture, software engineering and next-generation computing.

Dr. Seo-Young Noh received the BE and ME degrees in computer engineering from Chungbuk National University in Korea, and the MS and PhD degrees in computer science from Iowa State University, respectively. He is the director of the Center for Open Source Software in the National Program for Excellence in SW at Chungbuk National University. Before joining Chungbuk National University, he worked for the division of National Supercomputing at Korea Institute of Science and Technology Information, and LG Electronics as a principal researcher and a chief research engineer, respectively. His research interests are including scientific data management, cloud & scientific computing, Linux platforms and database management systems.