A Review of Recent Trends in Virtualization and its Impact

[1]Ganeshprasad Biradar, [2]Aditya Giridharan, [3]Chethan K P, [4]Yashwanth B K and [5]Smriti Srivastava
[1, 2, 3, 4] Undergraduate Student, [5]Assistant Professor
[1, 2, 3, 4] Computer Science and Engineering,
[1, 2, 3, 4] R.V. College of Engineering, Bengaluru, India

Abstract: In the world of computing, the process of using a virtual version of something, including but not limited to device storage, hardware platforms and network resources is known as virtualization. Applications of virtualization are many, ranging from Network Virtualization, to Desktop Virtualization, Application Virtualization, Data Virtualization, and Storage Virtualization. A majority of organizations these days make use of virtualization to converge their data center infrastructure and take advantage of this converged infrastructure. In this paper we survey different state of the art papers and current papers on virtualization, and we examine its impact on the world of technology.

Index Terms **- Virtualization, Abstraction, Cloud, Networking.**

I. Introduction

Virtualization refers to the abstraction of computer resources, be it processing power, storage or network resources. In the field of IT, before the period when virtualization became mainstream, the management of computer resources, user profiles and provisioning of developer tools was tightly coupled with the support that Microsoft offered as Windows was the go-to operating system in enterprise environments. This resulted in high costs, less flexibility and less security for the developer environment. The cost of infrastructure associated with virtualization of desktops and applications and the sub-par user experience that it offered in the earlier days created a barrier for organizations to justify its implementation. With the advent of cloud technologies and advances in content delivery, virtualization finds an abundance of applications across many fields. In the IT environment, management of users and computer resources is made easier and more secure. A good end user experience is provided with the help of high end graphics processors deployed in data centers. The end user can access high performance applications and desktop environments over thin clients like Chromebooks and tablets offering the convenience of mobility of any environment.

Virtualization is also affecting other networking devices like Application Delivery Controllers (ADC). These were traditionally deployed on premise hardware, which makes installing and maintaining them hard. Due to recent growth of virtualization and availability of public cloud providers like Amazon Web Services, Google Cloud Platform, it is very simple to deploy and configure application delivery controllers on cloud. This also makes it very easy for customers to try out different combinations of cloud providers and different application delivery controllers (for e.g. Citrix's ADC) available in their marketplace and then choose the best one among them.

II. Literature Review

The field of virtualization has been the subject of ongoing research since the early 1960s, when IBM began working on the CP-40, a predecessor of CP-67. CP-67 was a part of CP/CMS (Cambridge Monitor System), IBM's revolutionary virtual memory time-sharing operating system. Since then, the meaning of the term “Virtualization” has broadened and now applies to networks, desktops, applications, storage and data to name a few.

A lot of research has been dedicated to desktop virtualization. A large number of companies are choosing to converge their data-centers and provision systems on these data-centers for their employees to use, rather than invest in a large number of on-premise machines. This has allowed data and resources to become accessible from anywhere and has resulted in an increase in productivity.

Pranit Patil and Shakti Shekar [1] present an overview of the different implementation techniques for desktop virtualization and identify the best use-cases for each technique. They conclude that a Centralized Virtual Desktop Infrastructure is ideal for a scenario where the same set of applications has to be consistently delivered to a group of workers. On the other hand, client-side hypervisors are suitable when users are sophisticated and can work on two different operating systems and switch between them. This gives users a lot more flexibility. While both these methods are highly contrasting, the method of application isolation and workspace isolation offers a combination of both these methods and is likely to appeal to a much wider user base.

An important aspect of desktop virtualization is the ability to deliver an optimized performance to enhance user experience. A virtual desktop session must feel similar to a native session. Wang *et al* [2] carried out a study on optimizing interactive performance in different ways. Specifically, the study discusses performance optimization of virtual machines in a multi-core environment. They found that static binding between the virtual CPU and the core along with usage of QEMU (an open-source emulator that is capable of performing hardware virtualization) can greatly improve load balancing and decrease cache miss rate. A mix of VCPU and QEMU scheduling can be employed too.

Another paper that studied the quality of experience of user sessions on Remote Virtual Desktops (RVD) is [3], by Pedro Casas *et al*. The authors of this paper conducted extensive tests based on 52 different users' Remote Virtual Desktop sessions. It was found that comparing RVD services and native desktop applications is difficult over slow connections with high Round-Trip time and low down-link bandwidth. Nonetheless, when RVD services are provisioned over faster connections, quality of experience does improve. This also applies to mobile technologies like LTE (Long Term Evolution) and HSPA+ (High Speed Packet Access).

Hwang and Wood, in their paper on Adaptive dynamic priority scheduling for virtual desktop infrastructures [4] discuss a dynamic scheduling technique that could increase resource utilization when a small number of machines are used by a large group of users. They explain that the D-PriDe scheduler has a great effect in VM interference, bringing it down from **66%** to nearly **2%.**

A lot of research has also validated the impact of implementing desktop virtualization in various scenarios. For example, a study carried out by Sooyoung Yoo *et al* into the implementation of cloud based desktop virtualization at a hospital [5] showed that implementing a virtual desktop infrastructure (VDI) at SNUBH (Seoul National University Bundang Hospital) provided a return-on-investment of **122.6%** when 400 virtual machines were used, considering a five-year period. It also mentions the findings of previous studies in similar domains - reports ranged from **72%** electricity savings, to **66%** reduction in user account management time, lower costs of hardware, and even improved security when Thin Clients were used. Yet another paper written by Pawel Chrobak from the Wroclaw University of Economics highlights the impact of using a VDI in academic labs [6]. It concludes that using a VDI vastly improves the use of resources within the University and also provides convenience by allowing students to access software from any device or location. Administrative and maintenance costs are also greatly reduced.

In [7], Dorobont *et al*. explore the performance of remote virtualized desktop services against three main parameters - memory management, CPU usage and storage management. The virtualization services that are evaluated are two Microsoft remote desktop offerings - Microsoft Hyper-V and Microsoft Remote Desktop Services (RDS). Microsoft Hyper-V is a hypervisor that allows guest operating Systems (OS) to run on virtual machines (VMs) that share resources of the host and provision these VMs to remote thin clients. RDS delivers the host OS itself to remote clients. Both these services rely on Remote Desktop Protocol (RDP) for remote desktop delivery. The authors comment about the cost effectiveness of using virtualized desktop deployment on a central server station and accessing the VMs or the host OS over thin clients like single chip computers and smartphones. These services were deployed on a HP ProLiant DL 380p workstation with Intel Xeon E5-2609 v2 (4 cores) running Windows Server 2012 R2 and performance evaluation is done for the same. The results of the evaluation show that RDS outperforms Hyper-V in memory, CPU and storage management. The higher performance offered by RDS can be attributed to the fact that it is not associated with managing VMs over the host OS, unlike Hyper-V. CPU and memory management offered by both services are close but the storage management varies by a size-able margin. Disk queue length is observed as the indicator to evaluate storage management, and the authors observed a difference of orders of magnitude in the disk queue lengths between RDS and Hyper-V when stress tests were run. The authors concluded that even though the maintenance and acquisition costs of RDS is higher, the cost is justified by the better performance that it offers against Hyper-V.

In [8], the authors compare RDP and PCoIP (PC over IP) protocols across a various number of use cases. RDP is in-built within Windows OS whereas PCoIP is a protocol used by VMWare that offers better graphics capabilities. The focus of this paper is to examine whether the extra cost associated with licensing for the VMWare PCoIP (or any proprietary presentation protocol for that matter) is justified by the advantages it offers. Presentation protocols define the user experience that the end user gets, and support for protocols which offer a good user experience is important to establish easier adoption of desktop virtualization. The authors use a benchmark tool to simulate daily tasks and monitor certain performance parameters and use network monitoring tools to monitor bandwidth consumed by each of these protocols. The authors evaluate the protocols' ability to transfer a Windows 7 SP2 desktop session to a remote thin client by executing a set of generic tasks like Microsoft Office applications, Adobe Photoshop, web browsing and light games. It was observed that RDP offered better performance that PCoIP for almost half the use cases, getting higher scores than PCoIP for almost all benchmark suites. Network monitoring showed that RDP is also more efficient in terms of bandwidth utilization. The only case where RDP falls behind is graphics and multimedia. It was observed that RDP offered poor graphics compared to PCoIP, especially during video playback where stutters and artifacts were observed. The authors conclude that RDP is outperforms PCoIP in daily user tasks and the extra cost associated with PCoIP is not worth the gains in performance offered, if any.

In [9], the authors give a general overview of the types of virtualization of computer resources and also state the importance of virtualization in cloud computing. The authors review in simple terms, the computer resources that can be virtualized, briefly state the benefits of virtualization and the architecture of virtualization. An overview of the techniques used in virtualization is also provided, namely, full virtualization using binary translation, para-virtualization and hardware assisted virtualization. Virtualization is not a must to deploy cloud but it makes scalability, availability security and efficient usage of resources much easier to implement. The importance of virtualization in abstraction of the resource pool is emphasized in order to provide the mentioned features. The authors conclude that virtualization is becoming an important part of IT management to maximize resource utilization, provide users a way to access their workspace at any place and time in a secure manner and to add the convenience of policy management for administrators.

In [10], the problems and available solutions for executing remote graphics intensive tasks are explored. The authors give a comprehensive idea about the complexity associated with implementing graphics virtualization and graphics remote access. There is a lack of open-source software that support GPU virtualization to perform remote graphics intensive tasks. Currently, only a few vendors like Microsoft, Citrix and VMWare give complete implementations that offer remote graphics processing services, which are tightly coupled with hardware, OS and proprietary protocols. The paper elaborates on the different types of workloads that a graphics processing environment may come across. These are classified into graphics-agnostic/graphics-intensive workloads and content-oriented/computation-oriented workloads. The resulting four kinds of workloads have different requirements for graphics processing power. The workloads that need graphics processing power have been stated to have two main requirements; the first one being GPU sharing where GPU hardware is exposed to multiple guest OS or multiple software graphics servers within the same OS; the second one being specific server software (virtual graphics server - VGS) that provisions an interface to the GPU and client software (virtual graphics client - VCS that lies in the thin client being used. Both of these requirements have to be met to successfully implement a multi-user remote graphics environment. A note is made on the vendors offering remote workspace solutions. Microsoft offers RDP which is suitable for graphics-agnostic remote tasks and offers no support for GPU virtualization and therefore converts all GPU/3D tasks to CPU-intensive tasks. The hypervisor offered by Microsoft Windows Server OS does not support graphics virtualization either, requiring API interception (through RemoteFX) to use GPUs for remote tasks, sacrificing performance. Other vendors like Citrix (XenApp/XenDesktop with Citrix Receiver) and VMWare offer complete solutions for handling graphics intensive workloads. Several hardware configurations are explored by the authors to examine the possibility of desktop grade GPU performance with GPU sharing and observe that the best attainable GPU performance over a remote session reduces only by ~0.5% when compared to the actual GPU performance.

It is clear that Desktop Virtualization can be leveraged to greatly improve a lot of administrative and managerial tasks.

Network Virtualization is another area of virtualization with tremendous impact, and a lot of research.

In [11] the authors address the major existing gap between hosts, which in many cases are servers installed in a warehouse, and *service-centric* applications i.e. applications don't care if they are being hosted on one server or multiple servers. Because of this, the application service providers like Google, Amazon, and Microsoft employ a variety of techniques like data-plane proxying to dynamically allocate requests to one of many available hosts. However, since these data-centers are usually spread across regions, orchestrating requests at this large scale is very difficult even for the largest application service providers. The authors propose a new networking architecture called OpenADN which would allow the existing small scale application service providers to fully utilize the capabilities of cloud computing. This networking infrastructure is implemented using the idea of Software Defined Networking.

Ferdman, Ilia, *et al* [12] discuss the techniques and advantages of using virtualized application delivery controllers. Usually a hypervisor is used to manage multiple virtual machines. The hypervisor is the main point of communication between the virtual machines and the underlying hardware and its main job is to allocate resources dynamically to the virtual machines on demand. To do this, the hypervisor usually intercepts system calls made by the virtual machines and assigns this computation to free resources and then communicates the result back to the virtual machine. It essentially acts as layer of abstraction between the available hardware and the deployed virtual machines (VMs). But hypervisors are usually optimized for ubiquitous computing devices like servers and not networking devices like ADCs. The authors enclose a new virtualization solution which utilizes blade servers where each blade server has different necessary hardware parts like at least one network interface, multiple core processors, a memory and is configured to handle multiple virtual application delivery controllers (vADCs). Each of these vADCs will have access to resources of the particular blade server they are spawned on. The authors claim that this proposed architecture will provide more throughput when compared to the traditional solution of using a conventional hypervisor to deploy ADCs.

[13] talks about the most common challenges faced in implementing virtualization and proposes a new virtual switch called Open vSwitch. Traditionally, virtualization has been built around standard internet switching with virtual network switches in hypervisors which provide the logical connection between the virtual network interfaces and the physical network interfaces. But these conventional approaches are no longer sufficient to handle new features scaling across multiple data centers, migration of VMs between hosts and isolation of VMs in a multi-tenant solution where multiple customers have their VMs deployed on same hardware. According to the authors, the existing virtual switches are not making use of extra available information which the hypervisor has access to. Their proposed software, Open vSwitch, provides new interfaces to manage configurations and forwarding states at run time which makes supporting virtual environments easier. To change the existing configurations, the user has to provide key/value pairs which are then set by Open vSwitch. The user can also set up events which could be triggered based on configuration state changes. Forwarding tables can also be modified directly, which are a set of rules that tell how packets are handled based on their L2, L3 and L4 headers.

In [14] the authors discuss about the energy efficiency that can be achieved using server virtualization techniques. A lot of energy is wasted when servers are left idle in data centers. Hence by running multiple applications on a single server by virtualization and turning of the idle servers can result in a significant reduction in energy consumption. But virtual servers consume more energy that physical servers. So a count of the number of processes to be run on the virtual server and the hypervisor used must be considered to compare the energy consumption with an equivalent physical server. The experiment was conducted on three server which hosted CentOS as both host and guest and different hypervisors were used. The findings of the experiment was that the servers consumed same energy when in *off* state, in *on* state but in idle conditions the virtual server's CPU utilization was 0.5% to 2.3% more than the physical server. When in running conditions energy savings can be achieved by deploying appropriate number of Virtual Machines and it is found that energy consumption in virtualized servers with three VMs consume more energy than two VMs. It is experimentally found that around 15%-20% of energy can be conserved by switching the job from one VM to another and turning of the unused VM.

In [15] the authors discuss about the development of network function virtualization (NFV) using SDN techniques. A VM deployed on cloud can act as a high throughput platform thereby reducing the cost for business organization and more preferable than the standalone devices. But there are few limitations that limit the types of network functionality that can be virtualized. NFV provides solution to this problem by providing a software based data plane that can be run on VM. Intel's DPDK a software framework which acts as a fast packet processor allows applications to directly receive the packets from the NICs but this framework lacks few options to support virtualization. The authors have developed a platform called NetVM to improve this situation. NetVM is a platform for running complex network functionality using commodity hardware. It abstracts the DPDK capabilities and enable in-network services to be easily created on them. NetVM also allows grouping of servers which enables the safe multiplexing of network functionality from competing users. The comparison between the NetVM and the SR-IOV technique showed the NetVM has a better latency i.e. around 50μs whereas the latency in SR-IOV is around 83μs at line rate of 10Gbps. It is also possible to obtain complex network functionality using multiple VMs and still obtain a line rate which is nearly 250% more than the one which can be obtained with SR-IOV techniques.

Using NFV reduces hardware equipment cost but it also introduces various security challenges. In [16] the authors discuss about the NFV security challenges and provides solution to those problems. The security issue arises mainly due to the NFV infrastructure design and implementation faults. The challenges can be broadly categorized into three domains i.e., *hypervisor domain* where the unauthorized access and data leakage are the main concerns, *Compute domain* where the shared CPU and memory are under threat from external access and *Network domain* where the shared logical and physical layers can be compromised by external attacks. The authors provide solution to these problems by describing several techniques like usage of Policy manager to NFV, a user-centric NFV model where security is ensured by trusted third party domain in the same network, a control plane architecture called OpenNF to provide safe data flows across different instances in the network and several other techniques developed by Cisco, VMware and Alcatel-lucent which mainly deals with anomaly detection, event prediction and usage of OpenStack for QoS, Placement of VMs and formation of secure NFV cloud system.

Finally, [17] brings out some common security implications of virtualization. The tight integration of software and hardware and the combination of virtual networking, virtual routing and the usage of multiple host OS presents its own challenges. It stresses on the fact that the idea of implicit trust, i.e. the trust that the OS places on the virtual hardware and the VMM (Virtual Machine Monitor) can result in the VMM becoming a single point of failure. Snapshots and cloning of VMs also pose a significant threat of their own. The paper concludes by acknowledging the potential security pitfalls of virtualization and stating that care must be taken whole deploying, monitoring and managing virtual solutions.

III. Conclusion

It is evident that virtualization, as a technology, is transforming workspaces around the globe. By improving flexibility, allowing for greater security, and reducing costs to organizations, it has quickly gained traction and established itself. A lot of research has gone into optimizing various use cases of virtualization and bettering the overall experience. This purpose of this review paper was to highlight the trends in virtualization, with emphasis on desktop and network virtualization. It was found that there are ways to optimize performance and quality of experience using improvements in scheduling methods and CPU-core binding. In addition to this, comparisons have been carried out among different virtualized desktop services, which show objectively that RDS outperforms other services like Hyper-V. Similarly RDP, the protocol in-built into Windows OS outperforms protocols like PCoIP developed by VMWare. In the field of network virtualization, virtualized application delivery controllers and virtual switches have been suggested as improvements over traditional solutions. Network function virtualization is also discussed as being a solution to the limitations that hinder virtualization of network functionality. Workarounds, like OpenNF and Policy Manage, to security issues introduced by NFV are also discussed. Finally, a brief overview of the security challenges posed by virtualization was presented. As adoption grows for virtualization and the use-cases increase, more research will go into making it highly sustainable and efficient.

References

[1] Patil, Pranit. (2012). Desktop Virtualization Technologies and Implementation. International organization of Scientific Research. 10.9790/3021-0202310314.

[2] Wang X., Zhang B., Luo Y. (2013) Optimizing Interactive Performance for Desktop-Virtualization Environment. In: Zu Q., Hu B., Elçi A. (eds) Pervasive Computing and the Networked World. ICPCA/SWS 2012. Lecture Notes in Computer Science, Vol 7719. Springer, Berlin, Heidelberg

[3] P. Casas, M. Seufert, S. Egger and R. Schatz, "Quality of experience in remote virtual desktop services," 2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013), Ghent, 2013, pp. 1352-1357.

[4] Hwang J, Wood T. "Adaptive dynamic priority scheduling for virtual desktop infrastructures," In Proceedings of the IEEE 20th International Workshop on Quality of Service 2012 Jun 4, p. 16.

[5] Sooyoung Yoo, Seok Kim, Taeki Kim, Rong-Min Baek, Chang Suk Suh, Chin Youb Chung, and Hee Hwang. "Economic analysis of cloud-based desktop virtualization implementation at a hospital" BMC Med Inform Decis Mak, vol. 12, no. 1, 2012. Doi: 10.1186/1472-6947-12-119

[6] P. Chrobak, "Implementation of Virtual Desktop Infrastructure in academic laboratories," 2014 Federated Conference on Computer Science and Information Systems, Warsaw, 2014, pp. 1139-1146.
Doi: 10.15439/2014F213

[7] Örs Darabont, Konrád József Kiss , József Domokos, "Performance Analysis of Remote Desktop Virtualization based on Hyper-V versus Remote Desktop Services", International Conference on Recent Achievements in Mechatronics, Automation, Computer Science and Robotics, 2015, pp. 125-134.

[8] Casanova, L., Marcel, & Kristianto, E. (2017). "Comparing RDP and PcoIP protocols for desktop virtualization in VMware environment". 2017 5th International Conference on Cyber and IT Service Management (CITSM). doi:10.1109/citsm.2017.8089272

[9] Rakesh Kumar, Shipli Charu, "An Importance of Using Virtualization Technology in Cloud Computing", Global Journal of Computers & Technology, 2015, pp. 56-60.

[10] Vladimir A Smirnov, Evgeniy V Korolev and Olga I Poddaeva, "Cloud Environments with GPU Virtualization: Problems and Solutions", International Conference on Data Mining, Electronics and Information Technology, 2015, pp. 147-154.

[11] Jain, Raj, and Subharthi Paul. "Network virtualization and software defined networking for cloud computing: a survey." IEEE Communications Magazine 51.11 (2013): 24-31.

[12] Ferdman, Ilia, *et al*. "Techniques for virtualization of application delivery controllers." U.S. Patent No. 9,507,643. 29 Nov. 2016.

[13] Pfaff, Ben, *et al*. "Extending networking into the virtualization layer." Hotnets. 2009.

[14] Jin, Yichao, Yonggang Wen, and Qinghua Chen. "Energy efficiency and server virtualization in data centers: An empirical investigation." 2012 Proceedings IEEE INFOCOM Workshops. IEEE, 2012

[15] Hwang, Jinho, K. K Ramakrishnan, and Timothy Wood. "NetVM: High performance and flexible networking using virtualization on commodity platforms." IEEE Transactions on Network and Service Management 12, no. 1 (2015): 34-47.

[16] Yang, Wei, and Carol Fung. "A survey on security in network functions virtualization." 2016 IEEE NetSoft Conference and Workshops (NetSoft). IEEE, 2016.

[17] Pearce, M., Zeadally, S., and Hunt, R. 2013. "Virtualization: Issues, security threats, and solutions." ACM Comput. Surv. 45, 2, Article 17 (February 2013), 39 pages. DOI = 10.1145/2431211.2431216