Abstract

Society's reliance on the Internet has seen explosive growth over the past two decades. This reliance has become more evident in the last few years with the rise in affordable mobile Internet, and more recently due to COVID-19, which has caused various activities to be performed over the Internet on a mass scale. This brings forth some interesting questions; how prepared we are to support such mass-scale online activities and how well our current infrastructure is suited. Data centers handle the computational power required by such actions. Data centers have been developed and used over the past 50 years; however, the workload they handle has evolved. Consequently, enterprises have been looking at alternative architectures, which can support such changing workloads. One such alternative is hyperconverged architecture that is considered to be the next-generation architecture for modern data centers. An interesting difference between traditional data center architectures and hyperconverged architecture is how it handles storage and interacts with the existing network traffic since a common channel is shared between them. This thesis tries to shed some light on the nature of the interaction of the storage and the network traffic over hyperconverged data centers, the possible issues that can occur when different types of workloads intermingle, and the methodology to solve them.

The first contribution of the thesis explores how different types of workloads interact in a hyperconverged data center and how the fundamental operations, such as VM migrations, are affected by different workloads. We objectively show that disk-intensive operations are the most disruptive workloads, particularly with traditional network workloads, affecting other ongoing operations. We also demonstrate that this interference is inevitable when workloads utilize a shared communication channel, which is the case for hyperconverged data centers. Once we understand the impacts of interference among different workloads, we shift our focus to disk and network workloads. We argue that a solution for interfering workloads on the network layer should be handled at the network layer. This, however, is not a trivial problem since data centers are geo-distributed in nature, and it is hard to apply policies in a centralized manner. We, therefore, develop a solution, called *Hierarchical Two-Dimensional Queuing* (H2DQ), which relies on Software Defined Networking (SDN) that allows us to get a centralized view of the network. H2DQ enables us to create network policies dynamically for different types of traffic generated by the various workloads and provide them with resources in a fair manner.

However, system administrators of hyperconverged data centers prefer to have maximum control over their servers, including outgoing/incoming link throughput. A network-based solution might not always allow the management to reside with the system administrators. It is also possible to gather feedback regarding workload interference from the hypervisor instead of monitoring the network. With this

premise, we develop NetStor as the third contribution of the thesis, which tries to allocate resources for different workloads in a data center dynamically and, if required, migrates the workload to different locations. The approach shows a lot of promise compared to approaches in the existing literature.

Nevertheless, the design of NetStor is based on a virtualization standard – the virtual machine (VM). However, the industry has diversified and included more lightweight forms of virtualizations, such as containers. In the final contribution of the thesis, we develop a methodology called CONtrol, which is specifically geared towards data centers using containers. CONtrol gathers quality of service (QoS) metrics directly from the individual containers. It adjusts resources among other containers across multiple servers using a *Proportional Integral Derivative* (PID) controller to maintain ideal QoS. CONtrol's lightweight design is suited to handle a large number of containers in hyperconverged systems.

In this thesis we show how running mixed workloads in hyperconverged systems can cause major issues if it is allowed to run unchecked. However, by using our proposed resource management models such as NetStor and CONtrol, we can reduce application QoS drops by $5\% - 88\%$. We expect this thesis to discuss the importance of intelligently managing data centers and open avenues for new research in this domain.

Keywords: software defined, virtualization, data center, hyperconverged, container