Applied Cloud Computing and Big Data

PA2577 Quiz by Uday Jain

The Virtualization Reality: Are hypervisors the new foundation for system software?

Crosby, Simon, and David Brown

Virtualisation exploits the availability of high computing power and enables a single computer to host multiple different operating systems (OS) thereby evading the need to manage multiple or complex hardware infrastructure. The article summarizes the evolution of system virtualization through the years. Subsequently it deep dives into Xen hypervisor and its paravirtualization architecture compared against full virtualisation architecture followed by an overview of how it helps the evolving landscape of IT systems and the growing need to achieve economies of scale through shared resources such as CPU, memory, I/O, and a restricted views of devices across different OS systems running on the same hardware.

The article explains the role of a hypervisor or VMM (virtual machine monitor) to act as bridge between the host hardware and guest virtual machines (VMs), providing easily access to all hardware structures, such as the MMU (memory management unit), DMA (direct memory access) controllers and I/O devices, through application programming interfaces (APIs). These features help achieve benefits such as server consolidation, dynamic provisioning, high availability, fault tolerance as well as utility computing (where compute resources are dynamically shared based on availability and need). The benefits help pre-configuring most administrative tasks into a packaged VM that can be easily provisioned, highly portable and provides high degrees of security.

Hypervisors vs. lightweight virtualization: a performance comparison

Morabito, Roberto, Jimmy Kjällman, and Miika Komu

With growing adoption of cloud services that rely heavily on container based virtualizers, it becomes important to understand the performance trade-offs when using such solutions as well as other virtualizers. The paper evaluates non-virtualized solutions, hypervisor based virtualisation using Kernel-based virtual machines (KVM) and contained based virtualisation using Linux containers (LXC) and Docker containers. The paper also compares these solutions against hybrid hypervisors such as Operating System for Virtual Machines (OSv). A quick qualitative evaluation comparing the features and limitations for each of the solutions is done followed by a quantitative evaluation done based on performance of stack across CPU, Memory, Disk I/O, and Network I/O performance when running generic workloads. While Y-cruncher, NBENCH,noploop and Linpack are used to measure CPU performance, Bonnie++ s used for evaluating Disk I/O Performance across block output, block input and random write. STREAM is used for memory performance and Netperf is used for network I/O performance across TCP, UDP and TCP_RR protocols. With versatility and ease of management as value additions, the experiment concludes with container based hypervisors providing negligible overhead when compared with non-virtualized solutions. Due to limited development of OSv at the time of writing the paper, the author fails to provide conclusive comments but showcases promising results and scope especially given its relatively small size.

[Energy efficiency comparison of hypervisors]

Jiang, Congfeng, et al.

With the growing need for data centers providing cloud based virtualized machines, it becomes imperative to evaluate not only the performance but also energy efficiencies of available solutions to influence efficient design of data centers. Given the hardware abstraction and the semantic gap between the VM and the underlying hardware (host system), the VM OS cannot exploit power management available in non-virtualized environments. The paper provides a comprehensive evaluation of three orthogonal dimensions: hardware, hypervisor, and workload to help understand how to choose a hypervisor given host systems and workload needs?

The study was conducted for VMware ESXi, Microsoft Hyper-V, KVM, XenServer and Docker on 2 U rack servers, ARM64 server, desktop server and laptop using PrimeSearch for computation-intensive workload, STREAM for memory-intensive workload and LAMP for mixed workload. Varied intensity of workloads were evaluated. Since no single hypervisor was found energy efficient across all workloads or hardware options, the study concluded with key insights recommending different hypervisor solutions based on workload and available hardware to minimize energy consumption as well as maximize performance.

[A survey on the security of hypervisors in cloud computing]

Riddle, Andrew R., and Soon M. Chung.

The exponential increase in adoption of cloud based VM solutions resulted in a similar growth in malicious attacks on VMs to steal either the data from other VMs sharing hardware or more resources than allocated. The paper provides a literature review of different techniques such as side-channel attacks, memory and I/O based attacks, attacks on hypervisor using control data known as virtual machine escape, return oriented programming (ROP) attack on Hypervisors, non-control data attacks on hypervisors and VM rollback attack, commonly used to compromise the hypervisor. The paper also provides a quick overview of a few commonly used defenses against each of these attacks. Since attacks on hypervisor itself are the most dangerous, compromising the security of all hosted VMs, the discussion concludes with outlining different VM isolation techniques such as Secure Turtles, use of restricted hypervisor, use of a mandatory access control (MAC) policy to resourcing (even across hypervisor) along with a bind-time authorization and a Chinese-wall, and an architecture combining 2 hypervisors to providing a multi-level set of security requirements, to provide enhanced security of virtualisation solutions.

The review helps understand the vulnerabilities of the system and effort in the direction. It also establishes how some of the available solutions have performance trade-offs, impacting the system as a whole.

Short Questions:

1. What is a hypervisor?

A hypervisor is a virtual machine monitor (VMM) that creates a bridge between the host hardware and guest virtual machines (VMs), providing easily access to all hardware structures, such as the MMU (memory management unit), DMA (direct memory access) controllers and I/O devices, through application programming interfaces (APIs). Hardware abstraction by hypervisors allows multiple VMs with different OS to run on a single hardware sharing resources.

2. What are the main differences between a lightweight virtual machine and a virtual machine?

While a virtual machine runs its own OS on top of hypervisors that abstracts hardware and device drivers, lightweight virtual machines or containerised VM exploit the host OS and isolate processes at OS level, allowing only required processes to run within separate containers.

3. What is a microservice?

A microsservice is a fine grained, independent, small in size, autonomously developed, independently deployable, decentralized and simplistic process that can communicate with other microservices using lightweight generic protocols to stitch together a micro-service architecture based application capable of solving a business need.

4. What is the REST architecture pattern? How is it implemented in modern microservice development?

Representational State Transfer or REST architecture pattern outlines constraints and guidelines for development of and communication among web based stateless applications ensuring scalability, reliability, security, portability as well as performance.

Using REST principles such as using HTTP methods (GET, POST) to design APIs for microservice that return relevant error codes and is hosted using web based framework such as flask, we can implement a deployment on either an on-prem server or a cloud based server such as Azure, AWS or GCP.

5. What is "Infrastructure-As-A-Service" (IAAS)?

Cloud providers such as GCP, Azure and AWS today provide computational, storage and other necessary resources to deploy, host and manage applications without the need to manage the underlying hardware infrastructure. IAAS solution essentially allows developers to request infrastructure resources without worrying about the provisioning, management or scaling of underlying hardware. These tasks are administered by actors providing IAAS or cloud providers in this case. Consequently, when using IAAS, developers would need to worry only about the software components required for development.

6. What are the main differences between microservice development and IAAS?

Microservice development is to do with software layer and development of services that can perform certain simple tasks. However, IAAS is a service that manages underlying hardware components helping developers (of microservices for example) to provision storage, RAM or networking power on demand.