# Graduate School of Science and Technology Master's Thesis Abstract

| Laboratory name (Supervisor) | Software Design and Analysis (Hajimu Iida (Professor )) | | |
|---|---|---|---|
| Student ID | 2011414 | Submission date | 2022 / 7 / 21 |
| Name | LI GUOQING | | |
| Thesis title | The Convergence of Container and Traditional Virtualization: Strengths and Limitations | | |


Abstract

Virtual Machines (VMs) are used extensively in the cloud. The underlying hypervisors allow hardware resources to be split into multiple virtual units which enables server consolidation, fault containment and resource management. However, VMs with traditional architecture introduce heavy overhead and reduce application performance. Containers are becoming popular options for running applications, yet such a solution raises security concerns due to weaker isolation than VMs. We are at the point of container and traditional virtualization convergence where lightweight hypervisors are implemented and integrated into the container ecosystem in order to maximize the benefits of VM isolation and container performance. However, there has been no comprehensive comparison among different convergence architectures. To identify limitations and best fit use cases, we investigate the characteristics of Docker, Kata, gVisor, Firecracker and QEMU/KVM by measuring the performance of disk storage, main memory, CPU, network, system call, and startup time. On top of that, we evaluate their performance of running the Nginx web server and the MySQL database management system. We use QEMU/KVM as an example of running traditional VMs, Docker as the standard runc container, and the rest as the representatives of lightweight hypervisors. In addition, we cover the whole spectrum of virtualization technologies, starting from the first virtual memory implemented in IBM 370 system to a widely adopted container orchestration system Kubernetes. We study how virtualization is achieved by trap and emulate in the early days and how x86 ISA limitation is addressed using binary translation. To complete the picture of modern virtualization, we examine the underlying kernel features: cgroups and namespaces that enable process and resource isolation. we compare and analyze the benchmark results, discuss the possible implications, explain the trade-off each organization made, and elaborate on the pros and cons of each architecture. At the end of this dissertation, we include a case study of KubeVirt which is an extension of Kubernetes that enables running VMs along side with containers, we present our findings related to CPU pinning and NUMA pass-through and how we can achieve optimal performance with tuning.