Chapter 7 **Containerization**: Containers as a Service and Container Security

Swapna Mudrakola

b https://orcid.org/0000-0003-2816-6857 Matrusri Engineering College, India

Krishna Keerthi Chennam Vasavi College of Engineering, India

Shitharth Selvarajan https://orcid.org/0000-0002-4931-724X Leeds University, India

ABSTRACT

Applications are developed and deployed on the specific stack of software, as virtualization creates an environment to run the designed stack of software. Virtualization requires the installation of the entire software to run the application. The environment setup time will take more than, the execution time. Memory and CPU time are not effectively utilized. The containers are replacing the drawback of installing the unnecessary services of the software, not requiring running applications. The container consists of only the required services software loaded in the container. Container security is also an important aspect of utilizing applications with portability features. Different tools are designed for different aspects of security issues.

1. INTRODUCTION

Serverless computing in a cloud environment implements and runs small applications without interacting with the server regularly. The applications developed in serverless computing technology focus on developing an interactive and user-high-end technology that deploys on containers. The containers can

DOI: 10.4018/979-8-3693-1682-5.ch007

be run by the cloud service provider and manage services like operating systems, security management, monitoring systems, planning, etc as per Wen (2023).

Figure 1. Architecture environment for containers in serverless computing



1.1 Characteristics of Containers

Containers are the individual packages of resources required to run applications or services. Containers are used to deploy, run applications, and scale up services. Microservices run on containers, and containers provide the required environment to execute the application. The containers can host on Linux and specific Windows operating systems, and microservices can execute on developer workstations and also on local systems. Containers running locally will not cost, but server communication will cost only execution time and auto-scaling of services as per Ambrosino (2023). The operating system kernel is shared to run applications in containers, whereas, in virtual machines, the entire operating system is loaded to run applications. The containers can be deployed on bare metal servers or a cloud virtual machine. A single application can run multiple containers based on the required environment, like a web server, database, and application server, on different containers as per Wang (2023).

1.2 Benefits of Containerization

Containerization can run the application in different programming languages with a single operating system. It works on the principle of the microservices model. The task can be divided and run on multiple containers and operate clusters for orchestrators using Kubernetes as per Yang (2023).

Figure 2. Benefits of containerization



9 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/containerization/343723

Related Content

The Development of a Parallel Ray Launching Algorithm for Wireless Network Planning

Zhihua Lai, Nik Bessis, Guillaume De La Roche, Pierre Kuonen, Jie Zhangand Gordon J. Clapworthy (2013). *Development of Distributed Systems from Design to Application and Maintenance (pp. 69-86).* www.irma-international.org/chapter/development-parallel-ray-launching-algorithm/72247

Personal Storage Grid Architecture: Consuming Cloud Data Space Resources

Mian-Guan Lim, Sining Wu, Tomasz Simon, Md Rashidand Na Helian (2010). *International Journal of Grid and High Performance Computing (pp. 17-30).* www.irma-international.org/article/personal-storage-grid-architecture/45744

Rough Set-Based Feature Selection: A Review

Richard Jensen (2008). *Rough Computing: Theories, Technologies and Applications (pp. 70-107).* www.irma-international.org/chapter/rough-set-based-feature-selection/28468

QoS in Grid Computing

Zhihui Du, Zhili Cheng, Xiaoying Wangand Chuang Lin (2009). *Handbook of Research on Grid Technologies and Utility Computing: Concepts for Managing Large-Scale Applications (pp. 75-83).* www.irma-international.org/chapter/qos-grid-computing/20510

Paradigms for Effective Parallelization of Inherently Sequential Graph Algorithms on Multi-Core Architectures

Assefaw Gebremedhin, Mostofa Patwaryand Fredrik Manne (2021). *Handbook of Research on Methodologies and Applications of Supercomputing (pp. 74-95).* www.irma-international.org/chapter/paradigms-for-effective-parallelization-of-inherently-sequential-graph-algorithms-onmulti-core-architectures/273395