Secure Architecture for Virtual Machine to Container Migration in Cloud Computing

Dr. S.S. Manikandasaran¹ and S. Raja²
¹Associate Professor, Department of Computer Science, Christhu Raj College, Panjappur, Tiruchirappalli, Tamil Nadu, India.
²Research Scholar, Department of Computer Science, Christhu Raj College, Panjappur, Tiruchirappalli, Tamil Nadu, India.

Abstract. Cloud Computing is extension of virtualization concepts which can be included ‘As a Service’. It has extensive resource scaling and automation capability to integrate everything together in holistic view. In recent days, Cloud Computing has multiple enhancements and it is maturing with more flexibility but still security issues such as Confidentiality, Integrity and Availability remain exists and cloud adoption is happening drastically even though if issues persist. Virtual machine is building blocks for the application landscape and application architecture is monolithic. Container is the next level of abstraction layer in cloud computing. To migrate the existing virtual machine workload into container based will get into lot of security issues. In this paper, we propose secure architecture to migrate the virtual machine workload to containers in Cloud Computing align with security parameters. Current industry trends and challenges migrate the workload into cloud with server-less architecture and containerization.

1. Introduction

Cloud Computing is the underpinning platform which give priority to security. The entire information technology transformation is adapted from virtual machine technology to container strategy. Container is the extensive technology and is optimized piece of virtual machine which will do process isolation in either bare metal hardware machine or virtual machine. In this paper, cloud adoption takes place from virtual machine to container with strong commitment of security parameters which comprises confidentiality, integrity and availability. The application migration will be progressive since the process of cloud adoption should not be any impact of existing application transaction. The application migration from monolithic architecture to micro-service architecture required more effort with strong security around the wall. Monolithic application architecture defines that heritage method and tightly coupled. Micro-service architecture is a modern way of managing the application with full-fledged automation capability. The twelve-factor methodology can be applied to apps written in any programming language and use any combination of backing services such as database, queue, memory cache etc. in micro-service-based application [1]. Security is eminent factors when customers intend to migrate legacy application into micro-service architecture based.

This paper is organized as follows. Section 2 presents the Components of Cloud Computing and section 3 narrates the related research works. Section 4 defines the problem of the proposed work. Methodology is presented in Section 5 and Section 6 explains the proposed security architecture. Analysis of Proposed work is presented in Section 7. Finally, conclusion is given in Section 8.
2. Components of Cloud Computing

There are two types of components in Cloud Computing. They are Cloud Enabled, and Cloud Native which has defined in this section.

2.1 Cloud Enabled

Cloud Enabled is the application that was moved to cloud, but it was originally developed in a traditional data center. Some characteristics of the application had to be changed or customized for the cloud. On the other hand, it is an application that was developed with the cloud principles of multi-tenancy, elastic scaling and easy integration and administration in its design. In nutshell, cloud enabled environment is having semi-automation capability and full automation capability covered in the Infrastructure as a Service (IaaS).

2.2 Cloud Native

Cloud native is the term used for fully cloud based application which operate in global scale and it can be accessed anywhere even if the internet is not working. The application services and data have been replicated in multiple locations. It can be hosted anywhere either in private cloud or in public cloud or in hybrid cloud. In cloud native application, the application deployment and functionality changes happen rapidly which is not possible in legacy application. Often, it is called as continuous innovation which aligns agile methodology. Cloud Enabled and Cloud Native Components in Cloud Computing are presented in Figure 1.

![Figure 1: Cloud Enabled and Cloud Native Components in Cloud Computing](image)

3. Related Research Works

Security remains the biggest objection to cloud computing and the number one inhibitor to broad scale adoption. Information Technology leaders are expected to enable the business, innovate and do more for less and cloud computing presents this opportunity. However, IT departments are concerned with reduced visibility into cloud datacentre, less control over security policies, new and yet unknown threats facing shared environments and the complexity of demonstrating compliance. In traditional perimeter-based security, the population is divided into Trusted and Untrusted users. The network is used to create a demilitarized zone to keep trusted users behind the firewall and make the enterprise applications accessible behind the firewall. Likewise, this demilitarized zone keeps untrusted users out because they are not a part of the enterprise and untrusted.

Dennis et al., defined the term “Cloud Native” [2]. Cloud Native is the modern platform as a service which has scalability and operating in global scale. Its well-defined parallelism has been achieved in Cloud Computing to handle multiple users concurrently. The first transformation initiative was in the industry that to adopt infrastructure as a service in both on-premises and cloud with extensive capability
of automation to spin the instance in one click button. Micro-service is new design pattern in the cloud environment easy to manage, replicated, scaled, upgraded and deployed independently. In Micro-service application architecture, communication mechanism is little different than the heritage application architecture, it uses remote procedure call mechanisms and advanced message queueing protocol. Linux operating system provides complete encapsulation of process by the name of namespace. The author described beyond micro-service defined about serverless platform which operate based on event driven techniques. Most cloud service providers have offering which related to serverless platform. It is often called as “Fully Managed” platform.

Salman basset et al., discussed point of view container security [3]. The authors defined about benefits of using container in applications and same performance cannot expect as bare metal or virtual machine. Container have visibility of very limited resources rather than controlling entire operating system. Container has efficient way of handling application requests, simplified management, portability and less attack surface for the workloads. The author described about core components of container like namespace, control groups, Linux capabilities, secomp, Linux security modules, selinux, appArmour and user namespace. The author did not describe concrete security parameter that were defined and described which is related to container.

Ashif Khan defined the virtual machine to container transformation towards the serverless platform [4]. The author explained about building blocks of application agility which is container encapsulates operating system process that allow own private namespace and computations resource limits including memory and CPU. The author has clearly demonstrated about container orchestration capabilities. Containers can run on multiple platforms. In container clustering platform, Orchestration layer is responsible to maintain the cluster state which is important to have running operations. It has scheduling capability, backups of container, garbage collection, file consolidation, index rebuilds such as binning and instance affinity. Security is the most important factor in container orchestration, ensure high security standards and integrity of the deployed services. The key difference stated between physical platform and container platform. The author has defined the various stage of process in the micro-service architecture, Service registry is responsible to contain the network location and it is highly available to avoid single point of failure. This author has defined types of discovery method i.e. client-based discovery method able to determine the network location of available instance and load balancer to handle them. Server based discovery method the client makes a request service via a load balancer. The load balancer is responsible to query against service registry and router each request to an available instance. Continuous delivery and deployment is the development practice when code changes happen. Monitoring and governance has been differentiated between physical infrastructure and container platform. In the Container platform, Tools is responsible to provide the white box tracing logs, logging events to an approved logging store and monitoring container performance.

Kennedy et al., discussed about new approach for designing security as a service for cloud native based applications [5]. The author re-designed security architecture from traditional web application to cloud native based application. It has defined how security assessment has been taken care before continuous integration process kick in but did not express about counter measure if assessment failed. The author discussed about security requirements of the cloud native applications which application security, Network security and data security. The application refactoring is importance key things highlighted when converting from traditional application to cloud native patterns. The secure architecture implements and captured the charts and response time but did not mention about the counter measure.

Jeeva et al., discussed about techniques of security containers from distributed denial of service (DDoS) [6]. The authors defined about the virtualization and containers. It has been highlighted that container spawn techniques. When container is created virtual Ethernet, interface will be connected through bridge interface. Networking component plays vital role in the Denial of Service (DoS) attacks. The container is required root privilege to control the process and it can connect the host kernel. Hackers can easily have exploited using memory allocation techniques.
4. **Problem Definition**

In recent Cloud Computing transformation strategy, the end users are facing different challenges when converting their workload from monolithic architecture to micro-service architecture design patterns. There are multiple methods to convert from monolithic to micro-service like re-writing the code of entire application which will take more effort and another one is migrating the existing workload to micro-service patterns which will have more challenges in all the efforts. Security is still million-dollar question when comes to micro-service-based application in cloud.

5. **Methodology**

Cloud computing promotes the serverless architecture and infrastructure as a code (IaaS). The end user face challenges when they decided to migrate into micro-service architecture but still need to consider about security parameters. This paper proposes an architecture to migrate application from monolithic into micro-service previous work involves migrating the traditional workload from on-premises into cloud and enhances the current architecture into convert traditional application into modern architecture with cloud native capability.

![Migration Methodological Diagram](image-url)

**Figure 2. Migration Methodological Diagram**

Figure 2 Methodological diagram represents multiple stages of the migration. It has involved in converting the existing monolithic architecture to micro-service architecture among these phases. Discovery is minimum requirement factor for details analysis to understand the application landscape.
6. The Proposed Security Architecture
There is existing work involved migrating virtual machine to virtual machine migration into on-premises into Cloud data center [7]. In this paper, the proposed secure architecture is to migrate from monolithic architecture workload to micro-service architecture which comprises minimum refactoring of application code. Monolithic architecture is hosted in traditional infrastructure which is slow to change, complex hence fragile, difficult to scale, lack of resiliency and flexibility but Micro-service architecture have standard exposure to business functions for interaction and collaboration. Table 1 provides details steps on how to migrate from monolithic to microservice architecture.

This architecture is deployed on self-service, elastic, cloud computing infrastructure preferable containers, autoscaling and resiliency. The Micro-service architecture meets all requirements but required extra focus on security vulnerabilities, confidentiality, integrity and availability. Container images are basic building blocks and important element in software development lifecycle. The proposed architecture will prevent code vulnerabilities in container-based platforms since shared kernel architectures need to be accessed via standard configuration and container profiles. Container orchestration is responsible to provide both coarse grained and fine-grained access control for the container. Networking is key factor for container orchestration platform efficiently. Container created three networks host, bridge and none for communication. In this paper, container communication is explained that how IP address is gained by the container during entire life cycle. Service discovery is the factor application required to know the network location.

The proposed architecture meets the security requirements when converting the workload from on-premises to cloud computing as micro-service pattern. The container images and Host / Hypervisor underlying infrastructure must be hardened.
➢ The data volumes are shared across containers. It must be encrypted when containers mount the data volume during transaction.
➢ Port forwarding strictly handled since when container spawns, virtual network interface that will be created and network packet sniffing should be disabled.
➢ User access should be handled as non-privilege user to prevent from exploitation.
➢ Logs will be shipped properly during the container transaction.

![Figure 3](image-url) Secure Architecture for Virtual Machine to Container Migration

7. Analysis of Proposed work
The defined architecture will help customers to migrate monolithic application into micro-service architecture with the less amount code re-factoring. It mainly focuses to convert the application from tightly coupled environment to loosely coupled environment. The proposed architecture in the cloud computing environment intends to use non-privilege container since non-privilege container potentially unsafe container Unique identifier (UID) is mapped into container host root user id. The container image should be built from trusted registry. The architecture covers the Cgroup limitation which inherits from parent. The normal user who uses container can reasonably DoS host by running consistent number of fork process to generate Payload. Due to this unlimited payload, the host kernel will run out of memory. The user limitation inherited from parent and ulimit is tied to the name to kernel. The resource sharing possibly ID maps, common kernel UID, PID maps. With this consideration, this paper proposes architecture for migrating monolithic application into micro-service.

Our proposed architecture will have well defined east-west traffic in Container cluster. East-west traffic is Layer 2 network communication within the environment which will be connected to bridge interface in the container host operating system. Namespace is the isolation for the Linux namespaces and deeply described about isolation through system calls. The differentiation between container and virtual machine is how process map in operating system is shared since container shares same kernel space and easy to exploit if host is having vulnerabilities. This one of the main factor and will lead easily for potential exploitation if we don’t handle properly. All definition for the container will defined when we scan the source codebase environment. The data volume is to be encrypted when container mount the volume during the transaction. The proposed architecture is enforced to use AppArmor for enhancing security. As Continuous Integration and Continuous Delivery are the important factor to bring agility in the micro-service architecture, these properties are default in the proposed architecture.

8. Conclusion
Cloud Computing transformation journey is in next stage. The business users are facing many challenges in each stage of technology adoption and bringing more attention to the researchers. The cloud service providers created platform to design the new application but less focus on the existing workload migration and conversion into cloud without impacting the current environment. The proposed new
secure architecture for virtual machine to container migration involves multiple stages of discovery and defines the target cloud environment. This paper has detailed the migration of workload into fully automated environment in cloud computing. In Future, the proposed model will be implemented and achieved for better results. It will be more focused to do research on how code level security and data security will be carried out once the services converted into monolithic to microservices.

9. References
[1] Twelve Factor methods  https://12factor.net/ accessed on 05 October 2018.
[2] D. Gannon, R. Barga and N. Sundaresan 2018 Cloud-Native Applications IEEE Cloud Computing, 4, 5, 16-21.
[3] Salman Baset, Stefan Berger, James Bottomley, Canturk Isci, Nataraj Nagaratnam, Dimitrios Pendarakis, J. R. Rao, Gosia Steinder and Jayashree Ramanatham Docker and Container Security White Paper
[4] A. Khan Key Characteristics of a Container Orchestration Platform to Enable a Modern Application IEEE Cloud Computing 4 5 42-48
[5] K. A. Torkura, M. I. H. Sukmana, F. Cheng and C. Meinel 2017 Leveraging Cloud Native Design Patterns for Security-as-a-Service Applications IEEE International Conference on Smart Cloud (SmartCloud) New York 90-97.
[6] Manikandasaran S. S. and Raja S 2018 Security Architecture for multi-Tenant Cloud Migration, International Journal of Future Computer and Communication 7 2 42-45.
[7] Shu, Rui & Gu, Xiaohui & Enck, William. 2017 A Study of Security Vulnerabilities on Docker Hub. 269-280. 10.1145/3029806.3029832.
[8] IaaS, PaaS and SaaS – IBM Cloud service models. [Online]. Available: https://www.ibm.com/cloud/learn/iaas-paas-saas accessed on 05 October 2018
[9] L. Arockiam and S. Monikandan 2013 Data security and privacy in cloud storage using hybrid symmetric encryption algorithm International Journal of Advanced Research in Computer and Communication Engineering 2 8 3064-3070.
[10] S.S. Manikandasaran, K. Balaji and S. Raja 2018 Infrastructure Virtualization Security Architecture Specification for Private Cloud International Journal of Computer Sciences and Engineering 06 02 10-14.
[11] https://www.ibm.com/blogs/cloud-computing/2014/08/13/deploy-cloud-enabled-cloud-centric-application/.