QoS-Aware Secure Live Migration of Virtual Machines

Waseem Mandarawi, Andreas Fischer, Hermann de Meer
Chair of Computer Networks and
Computer Communications
University of Passau
Passau, Germany
Email: {Firstname.Lastname@uni-passau.de}

Eva Weishäupl
Department of Management Information Systems
University of Regensburg
Regensburg, Germany
Email: eva.weishaeupl@wiwi.uni-regensburg.de

***Abstract*—The live migration of Virtual Machines (VMs) is a key technology in server virtualization solutions used to deploy Infrastructure-as-a-Service (IaaS) clouds. This process, on one hand, increases the elasticity, fault tolerance, and maintainability in the virtual environment. On the other hand, it increases the security challenges in cloud environments, especially when the migration is performed between different data centers. Secure live migration mechanisms are required to keep the security requirements of both cloud customers and providers satisfied. These mechanisms are known to increase the migration downtime of the VMs, which plays a significant role in the compliance to Service Level Agreements (SLAs). This paper discusses the main threats caused by live migration and the main approaches for securing the migration. The requirements of a comprehensive Quality of Service (QoS)-aware secure live migration solution that keeps both security and QoS requirements satisfied are defined.**

Keywords*—*virtual machine; live migration; secure migration; quality of service; downtime; service level agreement;

I. Introduction

IaaS cloud environments provide elastic and demand-based computing resources and save the cost of deploying permanent servers with high capacities. For this reason, public and private IaaS clouds are increasingly used by many sectors to provide storage and computing capacity to the applications when needed. Server virtualization is the underlying technology that enables the cloud management systems to allocate VMs to the organization applications in private clouds and to the customers in public clouds. One of the key technologies in virtualization is the live migration that transfers a running VM to another host with as little service interruption as possible. Live migration supports the elasticity of the virtual environment and allows the continuity of services under many situations such as maintenance, faults, and attacks. It can also be utilized for improving the operating costs by consolidation of resources, or for load-balancing among the host systems. Unlike cold migration, live migration of a VM has to handle both cold state (e.g., the virtual hard disk, hardware configuration) and hot state (e.g., CPU state, RAM contents).

One of the main challenges in cloud environments is protecting the customer data from different threats imposed by virtualization. This challenge is even more critical when migration is used. The migrated data is often transferred without encryption, thereby creating different threats for the migrated VMs. The need for a secure live migration of VMs is particularly critical when the migration between different data centers over a Wide Area Network (WAN) is performed. However, any secure migration mechanism will cause some overhead on the migration process. The live migration is not supposed to interrupt the running services on the VM for a long time. It is, therefore, important to investigate the cost of security in terms of the performance penalty experienced when using a secure live migration mechanism.

Two main metrics are important to evaluate the efficiency of live migration approaches. The first metric is the total time required to perform the migration by the source and destination hosts. The second metric is the time period in which the services running in the VM are interrupted. This period is known as the downtime of the VM. The maximum accepted downtime of a VM mainly depends on the requirements of the user and services running in the VM. These requirements are defined in the SLAs between the cloud providers and customers. It is very critical for the cloud provider to keep the compliance to these SLAs when live migration is used, and at the same time, keep the user data protected during the migration. Non-compliance to SLAs will cause monetary and reputational losses to the cloud provider.

The downtime depends on many factors such as the memory size of the VM, network bandwidth and load, load on the host systems, and mainly the migration strategy used for migrating the memory pages of the VM. Two main strategies are the post-copy and the pre-copy migration. According to [1], the post-copy migration first pauses the VM and transfers its processor and devices state to the destination host, initially ignoring RAM contents. The VM is then resumed at the destination host. Whenever the VM tries to access missing memory pages, it is temporarily paused and the missing pages are requested from the source host. The VM is resumed once the missing pages have been received. In contrast, the pre-copy migration strategy includes 6 stages: the initialization, the reservation of resources at the destination host, the iterative pre-copy that first copies the entire RAM then iteratively transfers modified pages, stopping the VM and copying the modified pages after the last iteration, the confirmation by the destination that it has a consistent copy of the memory, and finally the activation of the VM on the destination host [2].

Several solutions for secure live migration were presented but with little or no consideration of its cost and its effects on the QoS provided by the VM. This paper focuses on a QoS-aware secure migration solution that integrates the decision making with the security measurements to keep both secure migration requirements and QoS requirements satisfied. The solution considers many factors before making the migration decision such as the estimated downtime of the migrated VM and the maximum allowed downtime according to the SLA. This paper introduces the topic and defines the requirements of such a solution. It also presents the main threats imposed by migration and the main mechanisms used for secure migration, as well as their influence on VM downtime.

The rest of the paper is organized as follows. Section 2 describes the main threats on both the migrated VM and the cloud environment caused by live migration. It also discusses the main mechanisms presented by researchers to handle these threats. In Section 3, the correlation between the downtime and the SLAs is discussed in addition to the possible cloud provider losses imposed by SLA violation. The architecture and requirements of a QoS-aware secure live migration solution are presented in Section 4. Section 5 describes an early prototype of a testbed that is being prepared for evaluating the solution. Section 6 concludes this paper.

II. Secure Live Migration

Secure live migration mainly aims to protect the VM from third party attacks during the migration process. Two issues are to be discussed here: potential threats during live migration, and approaches for a secure live migration.

A. Threats

Two main categories of threats are faced with live migration: the abuse of the migration process itself, and attacks on the benign VMs during migration. An example of the first category is compromising the management system and creating undesired migrations, leading to a denial of service attack on the VMs and the involved hosts. Another example is migrating a malicious VM from a malicious host to a benign host to perform VM-escape or side-channel attacks against the host and other VMs respectively. In the second category, the migrated VMs might be susceptible to many attacks such as man-in-the-middle, denial-of-service, and stack overflow attacks [3]. The migrated data such as kernel memory, application state, sensitive data such as passwords and keys etc., are usually transmitted without encryption. The data then can be sniffed or tampered easily during the migration, thus compromising integrity and confidentiality of the VM data [3]. These attacks can be either active attacks that manipulate the migrated data or passive attacks that eavesdrop on sensitive data such as passwords [4]. The most critical active attack is hijacking a benign VM (that is migrated from a benign host) by a malicious host. These threats are of particular concern if migration is performed between servers of different widely distributed data centers [5]. This discussion helps to define the main requirements of a secure live migration. First of all, the source and destination hosts should be trusted to avoid the migration of a malicious VM to a benign host or hijacking the VM. The second important issue is to ensure authorized access to the management interface to prevent malicious users from initiating undesired migrations. The third and the most critical issue is the integrity and confidentiality of the migrated data.

B. Solutions

Many researchers addressed the issue of secure live migration by defining the security requirements of the VMs, and proposing solutions and protocols that utilize existing security technologies to satisfy these requirements. Some researchers also evaluated the resulting downtime of the secure migration [6]. However, current research in the secure migration domain has not yet considered the QoS requirements of the VM and how to satisfy them during the migration. In general, many secure channel, trust establishment, and encryption mechanisms are used in these solutions. For example, the authors in [5], perform live migration over an IPsec (IP security) implemented transmission channel. This mechanism has a large overhead on the total migration time and the network traffic. The authors tried to decrease this overhead by adjusting the Maximum Transmission Unit (MTU) and Maximum Segment Size (MSS). IPsec and its protocols provide encryption, authentication and authorization while keeping the transmitted data integrity intact. The authors in [7] proposed a framework for a secure VM migration. The framework is based on hypervisors included in the Network Security Engines (NSE). It enables the traditional security approaches (such as firewalls, intrusion detection systems, and intrusion prevention systems) included in NSEs to work in the context of live migration. The framework transfers the security context along with migration data so that the VM can be restored at the destination.

The authors in [4] define a complete framework for secure live migration in which many mechanisms for satisfying the requirements of a secure live migration are used. The trust in the migration source and destination is ensured using hardware-based platform integrity verification using a Trusted Platform Module (TPM). The authorities of the administrator are ensured by defining role-based access control policies using the SUDO tool. The integrity and confidentiality of the migrated data are ensured using an SSH secure channel. Other security measures such as the firewalls and intrusion detection systems are deployed in the hosts.

TPM is widely used by researchers for secure migration. The authors in [8] also used TPM capabilities to design a trust credential for establishing trust with the migration destination platform. The authors in [9] also proposed a TPM-based trust protocol for migrating VMs between federated cloud providers. Other researchers used isolation for secure live migration. For example, the authors in [10] used an isolated migration network in which the source and destination hosts are grouped into a Virtual LAN (VLAN). Table I summarizes the mentioned solutions and which secure migration requirements (discussed in Section II-A) are considered by each solution.

III. VM Downtime and SLAs

Although live migration strategies are designed to have a small downtime, they still lead to a short service unavailability [11]. The downtime imposes significant economic aspects that need to be considered when live migration of VMs is used: service interruptions and unavailabilities lead to reduced business productivity and therefore to financial losses. In the worst case, service unavailability for a certain period of time might induce the loss of revenue since the customer's trust in

TABLE I. SECURE LIVE MIGRATION REQUIREMENTS CONSIDERED BY PRESENTED SOLUTIONS

Secure migration approach	Trust	Management authorities	Integrity and confidentiality
Bin Sulaiman et al. [5]	Yes	Yes	Yes
Xianqin et al. [7]	No	No	Yes
Anala et al. [4]	Yes	Yes	Yes
Aslam et al. [8]	Yes	No	No
Celesti et al. [9]	Yes	No	No
Anwar [10]	Yes	No	Yes

the service might be lost and he might not rely on the service provider in the future and switch to competitors instead. This will directly lead to a financial loss for the service provider. Moreover, reputation impact of a large downtime and regulatory compliances plays an important role. Nowadays, news about frequent or particularly long downtimes of a service are spread quickly and widely. In this case, the probability of long-term damage to an organization's reputation is higher than ever before and will lead to the loss of (prospective) customers. Furthermore, stringent government regulations require organizations to safeguard the reliability, privacy, and availability of customer data. Non-compliance with these regulations can lead to heavy financial penalties. In addition to governmental supervision, SLAs are part of most organization's contracts with customers or business partners. SLAs promise some level of service availability [11], leading to penalties if conditions are not fulfilled. If the service interruption during downtime exceeds the time determined in the SLAs so that the agreed-on level of service is not met, the firm could have financial penalties. According to the this argumentation, we assume that the monetary loss from downtime is calculated from the sum: "loss due to reduced business and productivity" + "loss due to losing customers" + "loss due to missing prospective customers" + "loss due to non-compliance of governmental and contractual requirements".

When live migration of VMs is used, the downtime needs to be minimized and financial consequences of the unavoidable downtime are imperative to be considered. Therefore it is essential to predict the worst case downtime as precisely as possible [11]. In the academic literature, the financial effects of downtime during live migration have not been sufficiently covered. Some researcher, however, studied the methods that might be used to estimate the downtime. The authors in [13] conducted a survey that summarizes the current approaches for evaluating the performance costs of VMs live migration. In [11], a model is presented that predicts the downtime based on a few characteristic parameters and is tested experimentally for applicability. The experiments in [14] have shown that live migration causes a longer than expected downtime, which results in a violation of SLAs.

IV. SOLUTION ARCHITECTURE

The security and QoS requirements imposed by live migration must be reflected by a comprehensive decision making process that considers all relevant environment parameters, user policies, and SLAs. The security parameters such as the threat level of the migration path, and security policies defined by the VM owner or the cloud service provider will be used to determine the required security measures during the migration. The performance parameters such as the network bandwidth and memory size of the VM will be used to estimate the migration cost. This cost will then be compared to the residual allowed cost determined by the SLA and downtime history of the VM. The migration is only allowed if the estimated cost does not violate the SLA. To achieve these goals, a central decision making module will interrupt any migration request issued in the cloud management system. This module will read the environment parameters, user policies, and SLAs to determine the required security mechanisms, estimate the migration cost, and finally make the correct decision. The decision making process is not yet integrated in any live migration or secure live migration solution neither in the academic literature nor in the commercial or open-source systems.

In this section, QSLM, a comprehensive solution for a QoS-aware secure live migration is described. Figure 1 depicts the proposed architecture. The central component of the solution is the QSLM algorithm that receives the migration request from the cloud management system and makes the appropriate decision. The implementation of this approach will develop add-ons for open source cloud solutions such as Proxmox [15] and Openstack [16] to intercept all migration actions. The decision of the algorithm is then translated to a set of actions and events that are communicated to the cloud management system and the cloud hosts. These actions include allowing or prohibiting the migration, and using certain security mechanisms if the migration is possible. The decision is taken according to the following inputs from the cloud management system:

- The security mechanisms deployed in the cloud hosts, also including an estimated overhead if possible. This information must be provided by the cloud administrators.
- The full topology of the cloud environment including the network links and bandwidth.
- The security metrics of the entities in the cloud environment. The simplest form of these metrics is the security of a communication link over which the migration will take place. For example, a wireless link might be considered to have low security. This information must also be provided by the cloud administrators.
- The security policies for each VM and cloud host. For example, a cloud customer might require all migrations of his VMs to be performed over secure channels. These policies are created by both the cloud system administrators and customers.
- The downtime budget for each VM. This value is determined from the SLA and the downtime history of the VM.

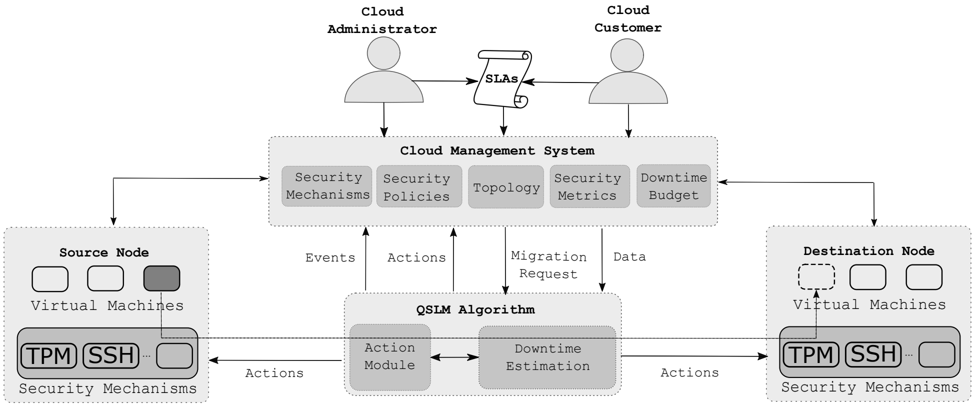

Fig. 1. QoS-aware Secure Live Migration (QSLM) Framework

In order to make the correct decision according to these inputs, the QSLM algorithm requires a module that estimates the maximum expected downtime of a VM depending on certain factors. The main factors are: the migration strategy, the size of the VM, the overhead of the security mechanism (e.g. encryption algorithm overhead), the network capacity and load, and the load in the source, destination, and the VM itself.

Some researchers addressed the estimation of the total migration time and downtime for different migration strategies (without considering the security mechanisms). For example, the authors in [12] defined a formula for estimating the downtime of the pre-copy migration strategy. The downtime of the VM includes only the last three stages. The main factor here is the number of the memory pages that are modified after the last copy iteration. This number depends on the activity of the VMs. In the worst case, all pages could be modified because of a heavy activity. The maximum downtime in this case is: $(\frac{VMSize}{LinkSpeed} + ConfirmationTime + ActivationTime)$, where $ConfirmationTime$ is the time required for the destination to confirm that it has a consistent copy of the memory, and $ActivationTime$ is the time required for the activation of the VM on the destination.

In QSLM, the downtime mathematical model will also include the overhead of the used security mechanisms. Since the encryption algorithm overhead is the main factor that increases the downtime, a mathematical model will be developed to estimate the expected execution time of different algorithms for a given CPU speed. The data overhead of the encryption algorithm will be also estimated and added to the VM size.

V. Evaluation Platform

The most important evaluation objective for the proposed solution is to measure the migration cost (mainly the downtime) caused by the live migration decisions made by QSLM and to compare this cost with the cost estimated by the solution according to the used security mechanisms and environment parameters. This section describes a simple testbed for evaluating the downtime of a VM that is migrated over a SSH channel in a WAN. The host system is a Debian-based Proxmox virtual environment. The testbed uses a Gigabit Ethernet and is configured as shown in Figure 2. A Proxmox router serves as WAN simulator and can add a certain delay. The VM has a memory size of 512 MB and shared storage is used for migration. Before the migration, a tunnel is created between a virtual bridge on the target host and a virtual bridge on source host. When the VM is migrated to Host B, it is connected to a SSH bridge as shown in Figure 3.

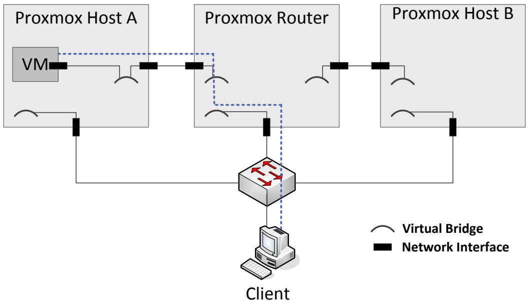

Fig. 2. Network topology before the migration

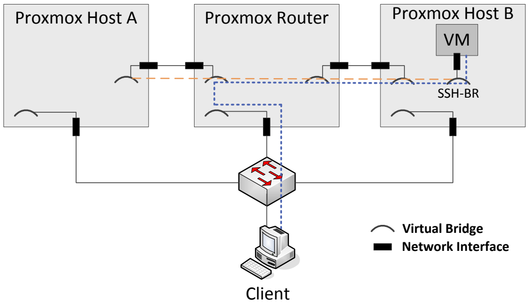

Fig. 3. Network topology after the migration

VM downtime is measured by running a program inside the VM that continuously sends packets to a client system. By measuring packet inter-arrival times the downtime of the VM can be estimated. Figure 4 shows the packets received at the client from the VM when migration is performed while the VM is in an idle state and without adding a delay in the router. The packets received before, during, and after the migration

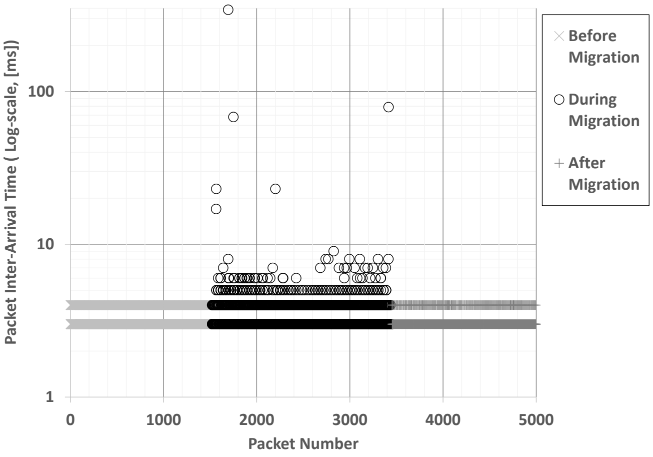

Fig. 4. Received packets at the client

are depicted with the packet inter-arrival times. The inter-arrival times indicate that the maximum downtimes during the migration is 342 milliseconds. The figure also shows a total migration time of about 7 seconds and larger inter-arrival times during the migration due to the load on the network. The future evaluation of the implemented solution will consider other security measures, realistic loads and memory sizes, delays in the migration path.

VI. Conclusion

The current approaches of secure live migration only address security requirements with limited performance cost measurements. Cloud service providers cannot easily adopt these solutions since the performance loss might violate the SLAs, especially for sensitive services. The downtime of a VM that is migrated over a secure channel is not trivial in realistic scenarios. Measuring this time even within large environments is not enough to provide full control of the process since the downtime is a dynamic factor that depends on the current situation of the VM and possibly the environment. These facts lead to the need for a comprehensive framework that can make the correct decision according to the many environment factors. A main requirement of this framework is a downtime estimation approach. Future work will focus on several directions: The solution will be implemented and evaluated. Different security mechanisms will be integrated and the overhead of these mechanism will be estimated. The measurement of the downtime will then be extended to realistic scenarios, possibly using cloud simulation solutions.

Acknowledgment

The research leading to these results was supported by the 'Bavarian State Ministry of Education, Science and Arts', as part of the FORSEC research association.

References

[1] T. Hirofuchi, H. Nakada, S. Itoh, and S. Sekiguchi, "Reactive consolidation of virtual machines enabled by postcopy live migration," in *Proceedings of the 5th International Workshop on Virtualization Technologies in Distributed Computing*, ser. VTDC '11. New York, NY, USA: ACM, 2011, pp. 11–18. [Online]. Available: http://doi.acm.org/10.1145/1996121.1996125

[2] D. S. Milojicic, F. Douglis, Y. Paindaveine, R. Wheeler, and S. Zhou, "Process migration," *ACM Comput. Surv.*, vol. 32, no. 3, pp. 241–299, Sep. 2000. [Online]. Available: http://doi.acm.org/10.1145/367701.367728

[3] M. Aiash, G. Mapp, and O. Gemikonakli, "Secure live virtual machines migration: Issues and solutions," in *Advanced Information Networking and Applications Workshops (WAINA), 2014 28th International Conference on*, May 2014, pp. 160–165.

[4] M. Anala, J. Shetty, and G. Shobha, "A framework for secure live migration of virtual machines," in *Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on*, Aug 2013, pp. 243–248.

[5] N. Bin Sulaiman and H. Masuda, "Evaluation of a secure live migration of virtual machines using ipsec implementation," in *Advanced Applied Informatics (IIAIAAI), 2014 IIAI 3rd International Conference on*, Aug 2014, pp. 687–693.

[6] A. Fischer, A. Fessi, G. Carle, and H. De Meer, "Wide-area virtual machine migration as resilience mechanism," in *Proc. of the International Workshop on Network Resilience: From Research to Practice (WNR 2011).* IEEE, Oct 2011, pp. 72–77.

[7] C. Xianqin, W. Han, W. Sumei, and L. Xiang, "Seamless virtual machine live migration on network security enhanced hypervisor," in *Broadband Network Multimedia Technology, 2009. IC-BNMT '09. 2nd IEEE International Conference on*, Oct 2009, pp. 847–853.

[8] M. Aslam, C. Gehrmann, and M. Bjorkman, "Security and trust preserving vm migrations in public clouds," in *Trust, Security and Privacy in Computing and Communications (TrustCom), 2012 IEEE 11th International Conference on*, June 2012, pp. 869–876.

[9] A. Celesti, A. Salici, M. Villari, and A. Puliafito, "A remote attestation approach for a secure virtual machine migration in federated cloud environments," in *Network Cloud Computing and Applications (NCCA), 2011 First International Symposium on*, Nov 2011, pp. 99–106.

[10] M. Anwar, "Virtual firewalling for migrating virtual machines in cloud computing," in *Information Communication Technologies (ICICT), 2013 5th International Conference on*, Dec 2013, pp. 1–11.

[11] F. Salfner, P. Tröger, and M. Richly, "Dependable estimation of downtime for virtual machine live migration," *Int. Journal On Advances in Systems and Measurements*, vol. 5, pp. 70–88, 2012.

[12] S. Akoush, R. Sohan, A. Rice, A. Moore, and A. Hopper, "Predicting the performance of virtual machine migration," in *IEEE Int. Symp. on Modeling, Analysis Simulation of Computer and Telecommunication Systems (MASCOTS)*, Aug 2010, pp. 37–46.

[13] A. Strunk, "Costs of virtual machine live migration: A survey," in *Services (SERVICES), 2012 IEEE Eighth World Congress on.* IEEE, 2012, pp. 323–329.

[14] W. Voorsluys, J. Broberg, S. Venugopal, and R. Buyya, "Cost of virtual machine live migration in clouds: A performance evaluation," in *1st Int. Conf. on Cloud Computing (CloudCom 2009).* Springer Berlin/Heidelberg, 2009, pp. 254–265.

[15] "Proxmox virtual environment," https://www.proxmox.com/en/proxmox-ve, accessed: 2015-07-22.

[16] "Openstack: Open source software for creating private and public clouds." https://www.openstack.org/, accessed: 2015-07-22.