American-Eurasian Journal of Scientific Research 11 (4): 265-270, 2016
ISSN 1818-6785

DOI: 10.5829/idosi.aejsr.2016.11.4.22903

An Extensive Survey on Co-Resident Attack in Dynamic Cloud Computing

[1]G. Nalinipriya, [1]P.J. Varalakshmi, [2]K.G. Maheswari and [3]R. Anita

[1]Department of IT, Saveetha Engineering College,
Anna University, Chennai, Tamil Nadu, India
[2]Department of MCA, Institute of Road and Transport Technology,
Anna University, Erode, Tamil Nadu, India
[3]Department of EEE, Institute of Road and Transport Technology,
Anna University, Erode, Tamil Nadu, India

Abstract: Cloud computing provides many advantages in scalability and cost efficiency, it also affected a number of security risks. This paper improved the co-resident attack, where malicious users aim to co-locate their virtual machines (VMs) on the same server and extract private information from the virtual machine. Most of the previous work has discussed hoe to mitigate the thread of co-resident attack. However the presented solution is impractical for the cloud platforms security problems. The problem from a different perspective and how to minimize the attacker's of co-locating their VMs in same server. Specially introduce security scheme policies of VM allocation policies. Our analysis shows the deploying three policies, the cloud provider decreases the attacker's possibility of achieving co-location by having a policy, where each policy is selected with a certain probability. These solutions do not require any changes to underlying infrastructure. Here it can be implemented in network analyzer tool and cloud sim tool.

Key words: *Co*-resident attack · VM policy · Co-locate · VM allocations · Malware attack · Vulnerability

INTRODUCTION

Virtualization has become an attractive in today's cloud computing environment. The ability to share the resources of a single physical machine server between several isolated virtual machines (VM) enabling a more optimized hardware utilization, as well as the easier management and migration of a virtual system compared to its physical counterpart, have given rise to new security polices and VM allocation plans. In particular, virtualization techniques is a key element in cloud computing. This paper gives an overview of some security problems related to the use of virtualization and shows that the widely used virtual machine managers cannot be considered fully secure. This paper presents first the virtualization principles and then discusses some vulnerability related to virtual systems before describing various attempts to address the security challenges raised by virtualization.

To address the security threats and issues relevant to cloud computing and virtualization [1], this guide outlines recommended security practices in virtual and cloud environments. For virtualized environments, public clouds, portions of hybrid clouds and public Infrastructure as a Service (IaaS) deployments, the enterprise, not the service provider, needs to assume responsibility for security.

Context
Definition
Efficiency: Virtualization not induce a significant decrease in performance. Therefore, the greatest amount of instructions must not require an intervention from the virtual machine manager (VMM).

Resource Control: The VMM must have a complete control over the virtualized resources.

Equivalence: A program must behave the same way on a virtual machine as it would do on its physical counterpart [2].

Corresponding Author: G. Nalinipriya, Department of IT, Saveetha Engineering College, Anna University, Chennai, Tamil Nadu, India.

Classification:	
Process Virtualization:	Virtualizing this layer consists in providing an interface between an application and the underlying system. This allows to create an application without concerning about the specificities of the OSes it will run on, as long as they possess the required virtualization layer. The Java Virtual Machine is an example of process virtualization.
Server virtualization	Here, the virtualization is applied to the hardware. This will allow many OS to run simultaneously on a physical machine. In this paper, we focus on server virtualization Techniques.
Network virtualization	VPNs (Virtual Private Networks, which enable the interconnection of distinct private networks through a public infrastructure such as the Internet) and VLANs (Virtual LANs, distinct local networks sharing the same physical infrastructure) are examples of network virtualization.
Storage virtualization	SANs (Storage Area Networks) fall into this category

Fig. 1: Classification of virtualization

Vulnerabilities and Attacks: Virtualization technologies offer new economical and technical possibilities. However, the addition of a new layer of software introduces new security concerns. Garfinkel and Rosenblum give in [3] a list of challenges raised by virtualisation that are discussed hereafter.

Scaling: Virtualization enables quick and easy creation of new virtual machines. Therefore, security policies of a network have to be flexible enough to handle a fast increase in the number of machines.

Transience: With virtualization, machines are often added to or removed from a network. This can hinder the attempts to stabilize it. For example, if a network gets infected by a worm, it will be harder to find precisely which machines were infected and clean them up when these machines exist only during brief periods of time on the network. Similarly, infected machines or still vulnerable ones can reappear after the infection was thought to be wiped out.

Software Lifecycle: The ability to restore a virtual machine into a previous state raises many security concerns. Indeed, previously patched vulnerabilities (programs flaws, deactivated services, older passwords. . .) may reappear. Moreover, restoring a virtual machine into a previous state can allow an attacker to replay some sequences, which renders obsolete any security protocol based on the state of the machine at a given time.

Diversity: In an organization where security policies are based on the homogeneity of the machines, virtualization increases the risk of having many versions of the same system at the same time on the network.

Mobility: A virtual machine is considered like any other file on a hard drive. It can be copied and moved to another disk or another host. This feature, cited as a benefit of virtualization, also adds security constraints because guaranteeing the security of a virtual machine becomes equivalent to guaranteeing the security of every host it has been on.

Identity: Usual methods used to identify machines (like MAC addresses) are not necessarily efficient with virtual machines. Moreover, mobility increases even more the difficulties to authenticate the owner of a virtual machine (as it can be copied or moved).

Data Lifetime: A hypervisor able to save the state of its VMs can counter the efforts made by a guest to delete sensitive data from its memory. Indeed, there may always be a backup version of the VM containing the data.

If many of these challenges can be addressed with good use policies (for cloud computing, examples can be found in [2], attackers may still exploit flaws in the system to perform their attacks. The rest of this section will focus on these malicious attempts to break into a virtualized system.

Virtualization and Cloud Computing Security Best Practices

Self-Defending VM Security: VM-level protection is crucial in a virtualized or cloud computing environment. By creating a security perimeter around each VM in this way, the enterprise can co-locate applications with different trust levels on the same host and can defend VMs in a shared, multi-tenant environment. This enables enterprises to maximize the benefits of virtualization, for example. And VM-level protection allows VMs to stay secure in today's dynamic data centers even as VMs travel between different environments – from on-premise virtual servers to private clouds to public clouds and even between cloud vendors.

Security Optimized for Virtualized and Cloud Environments: Security solutions should offer both agent-less and agent-based security options to provide flexible deployment alternatives and close security gaps unique to virtualized and cloud environments. Agent-less security is ideal for virtual infrastructures and private clouds. By leveraging hypervisor introspection application programming interfaces (APIs) such as the VMware VMsafe and vShield Endpoint APIs, businesses can now deploy a single antivirus engine to a dedicated security virtual appliance and deploy a very small footprint driver in each VM to perform the necessary off load. This provides the following advantages:

- Ensures other guest VMs are secure when dormant and receive the latest pattern file updates whenever activated[4-7].
- Enhances virtual server performance by running resource-intensive operations such as full system scans from the Separate scanning VM and staggering guest VM scans.
- Offers agent-less anti-malware, file integrity monitoring, host-based intrusion prevention, Web application protection, application control and firewall as agent-less security options. In a virtual environment, agent-less security uses the dedicated security VM to eliminate the agents from the guest VMs and reduce the resource burden on the underlying host – preserving performance and increasing VM densities. In a public cloud environment, businesses cannot use a dedicated scanning VM to protect other VMs because they do not control the hypervisor in a public cloud. Instead, an agent-based option provides protection on the VM level, creating self-defending VMs in a multi-tenant environment. Security solutions should provide both agent-less and agent-based options to protect across both virtual and cloud environments, all managed through a single console. With this approach, businesses can optimize virtual and cloud resources, simplify administration and reduce costs.

Protecting the VMs against their VMM: The purpose of CloudVisor [8] is to ensure data confidentiality and integrity for the VM, even if some elements of the virtualization system (hypervisor, management VM, another guest VM) are compromised. The idea is that data belonging to a VM but accessed by something else than this VM appears encrypted. To reach its goal, CloudVisor virtualizes the monitored hypervisor (realizing nested virtualization), therefore removing the latter from the most privileged zone while still giving it the illusion of the opposite. This means that the monitored VMM is now running in guest mode while CloudVisor is the only one in root mode. Any access, requested by the VMM, to some memory belonging to a VM is then trapped by Cloud- Visor. If the access is not requested by the owner of the requested page, CloudVisor encrypts its content. Such a concept seems particularly interesting within a cloud context, where multi-tenancy (unrelated users sharing the same physical resources) can be the norm and where privacy therefore represents one of the main concerns of cloud customers.

Attack Explaination

Denial-of-Service Attack: A System for Denial-of-Service Attack Detection Based on Multivariate Correlation Analysis [4]: Interconnected systems, such as Web servers, database servers, cloud computing servers and so on, are now under threads from network attackers. As one of most common and aggressive means, denial-of-service (DoS) attacks cause serious impact on these computing systems. In this paper, we present a DoS attack detection system that uses multivariate correlation analysis (MCA) for accurate network traffic characterization by extracting the geometrical correlations between network traffic features. Our MCA-based DoS attack detection system employs the principle of anomaly based detection in attack recognition. This makes our solution capable of detecting known and unknown DoS attacks effectively by learning the patterns of legitimate network traffic only. Furthermore, a triangle-area-based technique is proposed to enhance and to speed up the process of MCA. The effectiveness of our proposed detection system is evaluated using KDD Cup 99 data set and the influences of both non-normalized data and normalized data on the performance of the proposed detection system are examined[4]. The results show that our system outperforms two other previously developed state-of-the-art approaches in terms of detection accuracy.

Malware Detection: Malware Detection in Cloud Computing Infrastructures [5] Cloud services are prominent within the private, public and commercial domains. Many of these services are expected to be always on and have a critical nature; therefore, security and resilience are increasingly important aspects. In order to remain resilient, a cloud needs to possess the ability to react not only to known threats, but also to new

challenges that target cloud infrastructures. In this paper we introduce and discuss an online cloud anomaly detection approach, comprising dedicated detection components of our cloud resilience architecture. More specifically, we exhibit the applicability of novelty detection under the one-class support Vector Machine (SVM) formulation at the hypervisor level, through the utilisation of features gathered at the system and network levels of a cloud node. We demonstrate that our scheme can reach a high detection accuracy of over 90% whilst detecting various types of malware and DoS attacks. Furthermore, we evaluate the merits of considering not only system-level data, but also network-level data depending on the attack type. Finally, the paper shows that our approach to detection using dedicated monitoring components per VM is particularly applicable to cloud scenarios and leads to a flexible detection system capable of detecting new malware strains with no prior knowledge of their functionality or their underlying instructions.

Security Games for Virtual Machine: Security Games for Virtual Machine Allocation in Cloud Computing [6]. While cloud computing provides many advantages in accessibility, scalability and cost efficiency, it also introduces a number of new security risks. This paper concentrates on the co-resident attack, where malicious users aim to co-locate their virtual machines (VMs) with target VMs on the same physical server and then exploit side channels to extract private information from the victim. Most of the previous work has discussed how to eliminate or mitigate the threat of side channels. However, the presented solutions are impractical for the current commercial cloud platforms. We approach the problem from a dif-ferent perspective and study how to minimise the attacker's possibility of co-locating their VMs with the targets, while maintaining a satisfactory workload balance and low power consumption for the system. Specifically, we introduce a security game model to compare different VM allocation policies. Our analy-sis shows that rather than deploying one single policy, the cloud provider de-creases the attacker's possibility of achieving co-location by having a policy pool, where each policy is selected with a certain probability. Our solution does not require any changes to the underlying infrastructure. Hence, it can be easily implemented in existing cloud computing platforms [9-25].

Co-Resident Attacks in Cloud Computing: Virtual Machine Allocation Policies against Co-resident Attacks in Cloud Computing [7]. While the services-based model of cloud computing makes more and more IT resources available to a wider range of customers, the massive amount of data in cloud platforms is becoming a target for malicious users. Previous studies show that attackers can co-locate their virtual machines (VMs) with target VMs on the same server and obtain sensitive information from the victims using side channels. This paper investigates VM allocation policies and practical countermeasures against this novel kind of co-resident attack by developing a set of security metrics and a quantitative model. A security analysis of three VM allocation policies commonly used in existing cloud computing platforms reveals that the server's configuration, oversubscription and background traffic have a large impact on the ability to prevent attackers from co-locating with the targets. If the servers are properly configured and oversubscription is enabled, the best policy is to allocate new VMs to the server with the most VMs. Based on these results, a new strategy is introduced that effectively decreases the probability of attackers achieving co-residence. The proposed solution only requires minor changes to current allocation policies and hence can be easily integrated into existing cloud platforms to mitigate the threat of co-resident attacks.

Classifications of Tools:

Packet Analyzer or Network analyzer Tool	A packet analyzer (also known as a network analyzer, protocol analyzer for particular types of networks, an Ethernet sniffer or wireless sniffer) is a computer program or piece of computer hardware that can intercept and log traffic that passes over a digitals network or part of a network. As data streams flow across the network, the sniffer captures each packet and, if needed, decodes the packet's raw data, showing the values of various fields in the packet and analyzes its content according to the appropriate RFC or other specifications. Packet capture is the process of intercepting and logging traffic.
Cloud sim Tool	CloudSim goal is to provide a generalized and extensible simulation framework that enables modeling, simulation and experimentation of emerging Cloud computing infrastructures and application services, allowing its users to focus on specific system design issues that they want to investigate, without getting concerned about the low level details related to Cloud-based infrastructures and services.
Wire shark Tool	Wire shark is a free and open-source packet analyzer. It is used for network troubleshooting, analysis, software and communications protocol development and education. Originally named Ethereal, the project was renamed Wire shark in May 2006 due to trademark issues.

Fig. 2: Comparision of cloud Tools

CONCLUSION

Virtualization technologies can bring many interesting new features (optimized use of hardware resources, eased restoration, machine migration), but they also introduce new means to perform attacks. These attacks may either be directed against virtualized systems or leverage some features related to virtualization in order to take over a system. Therefore, especially if a system is shared between many users, as is the case in cloud computing, strong security of virtual machines is crucial to protect their data and gain the customer's trust. We have presented various implementations dealing with those concerns on two main topics: (i) Leveraging virtualization to secure a system located on top of a hypervisor.(ii) Securing the VMM. This also raised issues concerning the security of the most privileged element of a system. The research on these topics is very active today, following a great diversity of possibilities, going from devising ways to secure popular systems like Xen, KVM or VMWare solutions, to creating new models such as the microkernel-based virtualization. However, even if these solutions are satisfying security-wise, one should still consider the possible issues of their large-scale deployment. Indeed, additional control procedures will cause a decrease in efficiency. One must therefore find the right balance between performance and security, according to their needs.

REFERENCES

1. Virtualization and Cloud Computing Threat Report, Trend Micro. August 2011.
2. Alliance, C.S., 2011. Security guidance for critical areas of focus in cloud computing v3.0. Cloud Security Alliance.
3. Garfinkel, T. and M. Rosenblum, 2005. When virtual is harder than real: Security challenges in virtual machine based computing environments. In: Proceedings of the 10th conference on Hot Topics in Operating Systems, USENIX Association, 10: 20.
4. Zhiyuan Tan, Aruna Jamdagni and Xiangjian He, 2014. A System for Denial-of-Service Attack Detection Based on Multivariate Correlation Analysis, ieee Transactions on Parallel and Distributed Systems, 25(2).
5. Watson Michael, R., Noor-ul-hassan Shirazi, Angelos K. Marnerides andreas Mauthe and David Hutchison, Malware Detection in Cloud Computing Infrastructures, 10.1109/TDSC.2015.2457918, IEEE Transactions on Dependable and Secure Computing
6. Han Yi,Tansu Alpcan, Jeffrey Chan and Christopher Leckie, Security Games for Virtual Machine Allocation in Cloud Computing.
7. Han Yi, Jeffrey Chan, Tansu Alpcan, Christopher Leckie, 2014. Virtual Machine Allocation Policies against Co-resident Attacks in Cloud Computing, IEEE ICC, Communication and Information Systems Security Symposium.
8. Zhang, F., J. Chen, H. Chen and B. Zang, 2011. Cloudvisor: Retrofitting protection of virtual machines in multi-tenant cloud with nested virtualization. In: Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles. ACM, pp: 203-216.
9. Wang, Z. and X. Jiang, 2010. Hypersafe: A lightweight approach to provide lifetime hypervisor control-flow integrity. In: Security and Privacy (SP), 2010 IEEE Symposium on, IEEE (2010), pp: 380-395.
10. Yu, S., W. Zhou, W. Jia, S. Guo, Y. Xiang and F. Tang, 2012. Discriminating DDoS Attacks from Flash Crowds Using Flow Correlation Coefficient,IEEE Trans. Parallel and Distributed Systems, 23(6): 1073-1080.
11. Zhang, Y., A. Juels, M. Reiter and T. Ristenpart, 2012. Cross-VM Side Channels and Their Use to Extract Private Keys,Proc. 2012 ACM Conference on Computer and Communications Security - CCS '12, pp: 305-316.
12. Stillwell, M., F. Vivien and H. Casanova, 2012. Virtual machine resource allocation for service hosting on heterogeneous distributed platforms, in Proc. IEEE Int. Parallel Distrib. Process. Symp., Shanghai, China, pp: 786-797.
13. Klein, G., K. Elphinstone, G. Heiser, J. Andronick, D. Cock, P. Derrin, D. Elkaduwe, K. Engelhardt, R. Kolanski and M. Norrish, 2009. sel4: Formal verification of an os kernel. In: Proceedings of the ACM SIGOPS 22nd symposium on Operating Systems Principles, pp: 207-220.
14. Kortchinsky, K., 2009. Cloudburst—a vmware guest to host escape story. Black Hat USA.
15. Lacombe, E., V. Nicomette and Y. Deswarte, 2011. Enforcing kernel constraints by hardware-assisted virtualization. Journal in Computer Virology, 7(1): 1-21.
16. Zhang, Y., A. Juels, M. Reiter and T. Ristenpart, 2012. Cross-VM Side Channels and Their Use to Extract Private Keys. In: 2012 ACM Conference on Computer and Communications Security - CCS' 12: 305-316.

17. Aviram, A., S. Hu, B. Ford and R. Gummadi, 2010. Determinating Timing Channels in Compute Clouds. In: 2010 ACM Workshop on Cloud Computing Security Workshop – CCSW, 10: 103-108.
18. Vattikonda, B., S. Das and H. Shacham, 2011. Eliminating Fine Grained Timers in Xen. In: 3rd ACM Workshop on Cloud Computing Security Workshop - CCSW '11, 41-46.
16. Wu, J., L. Ding, Y. Lin, N. Min Allah and Y. Wang, 2012. XenPump: A New Method to Mitigate Timing Channel in Cloud Computing. In: 2012 IEEE Fifth International Conference on Cloud Computing, pp: 678-685.
20. Shi, J., J. Shi, X. Song, H. Chen and B. Zang, 2011. Limiting Cache-based Side-channel in Multi-tenant Cloud using Dynamic Page Coloring. In: 2011 IEEE/IFIP 41st International Conference on Dependable Systems and Networks Workshops (DSN-W), pp: 194-199.
21. Jin, S., J. Ahn, S. Cha and J. Huh, 2011. Architectural Support for Secure Virtualization under a Vulnerable Hypervisor. In: 44th Annual IEEE/ACM International Symposium on Microarchitecture - MICRO-44' 11: 272-283.
22. Wu, J., L. Ding, Y. Wang and W. Han, 2011. Identification and Evaluation of Sharing Memory Covert Timing Channel in Xen Virtual Machines. In: 2011 IEEE 4th International Conference on Cloud Computing, pp: 283-291.
23. Kadloor, S., S. Kadloor, N. Kiyavash and P. Venkitasubramaniam, 2012. Scheduling with Privacy Constraints. In: 2012 IEEE Information Theory Workshop, pp: 40-44.
24. Xia, Y., X. Yetian, Z. Xiaochao, Y. Lihong, P. Li and L. Jianhua, 2012. Constructing the On/Off Covert Channel on Xen. In: 2012 Eighth International Conference on Computational Intelligence and Security, pp: 568-572.
25. Bedi, H. and S. Shiva, 2012. Securing Cloud Infrastructure Against Co-Resident DoS Attacks Using Game Theoretic Defense Mechanisms. In: International Conference on Advances in Computing, Communications and Informatics - ICACCI' 12: 463-469.