ISSN:2348-2079

Volume-6 Issue-2

International Journal of Intellectual Advancements and Research in Engineering Computations

Attack Resistant Distributed Scheduling and Virtual Machine Management Framework for Clouds

[1]V. Sampath Kumar, [2]S. Vismeya, [3]M. Vivek, P. Kanmani, M.E (PhD),
Department of Computer Science and Engineering,
K.S.Rangasamy College of Technology, Tiruchengode

Email id: [3]mvivek3112@gmail.com

Abstract

Cloud computing environment supports high scalable hardware and software resource sharing platform through the Internet. The cloud provider shares the hardware resources with the cloud customers through the Virtual Machines (VM). Virtual Machines (VM) running on the same physical server are denoted as Co-resident VMs. The Co-resident Virtual Machines are logically isolated from each other. The logical isolation is violated by the side channels of the malicious users. The sensitive information from the Co-resident Vms are accessed by the malicious users is defined as Co-resident attacks. The Cryptographic keys, workloads and web traffic rates are the sensitive information accessed by the malicious users. The Co-resident attack is also referred as co-location, co-residence or co-residency attacks.

The Virtual Machine allocation policy is used to place the Virtual Machines in the physical server. The malicious user co-locates their VM to the target VM. The security, workload balance and power consumption parameters are considered in the Virtual Machine placement process. Secure metrics are defined to measure the safety of the VM allocation policy. The Balanced VM Allocation Policy is build to assign VMs in the physical servers. The Previous Selected Server First (PSSF) policy is used with security metrics. Least VM allocation policy, Most VM allocation policy and Random allocation policy are used with the workload balance parameter. The data centers are connected to the Virtual Machines with in the same environment.

The attack resistant Virtual Machine Management framework is build with centralized and distributed scheduling schemes. The live VM migration is protected from the side channel attacks. The system is enhanced with multiple data center management mechanism. The Distributed VM Placement (DVMP) policy is build to allocate the virtual machines on the physical server.

Index Terms: Cloud Resources, Virtual Machine Allocation Policies, Side Channel Attacks, Co-residential Attacks and Distributed Scheduling

1. Introduction

Public infrastructure-as-a-service (IaaS) clouds enable the increasingly realistic threat of malicious customers mounting side-channel attacks. An attacker obtains tenancy on the same physical server as a target, and then uses careful timing of shared hardware components to steal confidential data. Damaging attacks enable theft of cryptographic secrets by way of shared per-core CPU state such as L1 data and instruction caches, despite customers running within distinct virtual machines (VMs).

A general solution to prevent side-channel attacks is hard isolation: completely prevent sharing of particular sensitive resources. Such isolation can be obtained by avoiding multi-tenancy, new hardware that enforces cache isolation, cache coloring, or software systems such as StealthMem. Hard isolation reduces efficiency and raises costs because of stranded resources that are allocated to a virtual machine yet left unused. Another approach has been to prevent attacks by adding noise to the cache. For example, in the D¨uppel system the guest operating system protects against CPU cache side-channels by making spurious memory requests to obfuscate cache usage. This incurs overheads, and also requires users to identify the particular processes that should be protected.

A final approach has been to interfere with the ability to obtain accurate measurements of shared hardware by removing or obscuring time sources. This can be done by removing hardware timing sources, reducing the granularity of clocks exposed to guest VMs, allowing only deterministic computations, or using replication of VMs to normalize timing. These solutions either have significant overheads, as in the last solution, or

severely limit functionality for workloads that need accurate timing.

In addition to sharing resources and having access to fine-grained clocks, shared-core side-channel attacks also require the ability to measure the state of the cache frequently [10]. For example, Zhang et al.'s cross-VM attack on ElGamal preempted the victim every 16 μs on average. With less frequent interruptions, the attacker's view of how hardware state changes in response to a victim becomes obscured. Perhaps surprisingly, then, is the lack of any investigation of the relationship between CPU scheduling policies and side channel efficacy. In particular, scheduling may enable what we call soft isolation: limiting the frequency of potentially dangerous cross-VM interactions.

2. Related work

A variety of cross-VM side channels have been demonstrated in the academic literature. Deficiencies in performance isolation, similar to those leveraged in this work, have been exploited for a variety of purposes [7]. Noting that cache and network utilization are often contested between VMs, a resource freeing attack (RFA) has been proposed that allows a greedy customer to manipulate the performance of co-resident VMs by shifting their resource bottlenecks [4]. This work operates under a similar attack model as our own, targeting public network cloud services and manipulating VMs from a helper host. However, where RFA is a performance enhancement strategy for the cloud, co-resident watermarking is a method of information extraction.

Cache-based side channel attacks, in which timing differences in access latencies between the cache and main memory are exploited, have attracted the most attention for cloud computing. Most notably, Zhang et al. [6] demonstrated that the machine instructions of a co-resident VM can be recovered from shared L1 caches, permitting the reconstruction of secret keys in the circumstance that they influence the code path of a decryption routine. Ristenpart et al. showed that cache usage can be examined as a means to measure the activity of other instances co-resident with the attacker. Furthermore, they demonstrated that they can detect co-residency with a victim's instance if they have information about the instance's computational load. In contrast, Zhang et al. [5] utilized cache-based side channels as a defensive mechanism. Their scheme works by measuring cache footprints for evidence of other VMs. Leveraging this scheme, they can challenge correct functionality on the part of the cloud provider and discover other unanticipated instances sharing the same host.

Bowers et al. [8] have proposed use of a different network timing side channel in order to challenge fault tolerance guarantees in storage clouds. This work measures the response time of random data reads in order to confirm that a given file's storage redundancy meets expectations. This scheme can be used to detect drive-failure vulnerabilities and expose cloud provider negligence. We intend to investigate the applicability of storage cloud co-resident watermarking in future work.

Raj et al. proposed two other mechanisms for preventing cache based side channels, cache hierarchy aware core assignment, and page-coloring-based cache partitioning. The former groups CPU cores based on last level cache (LLC) organization and checks whether such organization has any conflict with the SLA of the clients. The latter is a software method that monitors how the physical memory used by applications maps to cache hardware, grouping applications accordingly to isolate clients. Another effective defense against cache-based side channels is changing how caches assign memory to applications, such as non-deterministic caches. Non-deterministic caches control the lifetime of cache items. By assigning a random decay interval to cache items, the cache behavior becomes nondeterministic, and hence, side channels cannot exploit it. Work in performance isolation in Xen can also lead to added security benefits.

Other work aims to combat virtualization vulnerabilities by reducing the role and size of the hypervisor. Most drastically, Keller et al. [2] eliminate a large attack surface by proposing the near elimination of the hypervisor. This is achieved through pre allocation of resources, limited virtualized I/O devices, and modified guest operating systems. While this approach inarguably reduces the likelihood of exploitable implementation flaws in the virtualization code base, it necessarily places VMs closer to underlying hardware. Intuitively, this can only increase the bandwidth of the isolation-compromising side channel that we explore in this work. Other proposals reduce the hypervisor attack surface by considering only specific virtualization applications such as rootkit detection or integrity assurance for critical portions of security-sensitive code [3] or by distributing administrative responsibilities across multiple VMs [10]. We do not consider these systems in our work because they are not intended for the third-party compute cloud model.

3. Virtual Machine Allocation Policies

Security is one of the major concerns against cloud computing. From the customer's perspective, migrating to the cloud means they are exposed to the additional risks brought about by the other tenants with whom they share the resources−are these neighbors trustworthy, or they

may compromise the integrity of others? This paper concentrates on one form of this security problem: the co-resident attack.

Virtual machines (VM) area commonly used resource in cloud computing environments. For cloud providers, VMs help increase the utilization rate of the underlying hardware platforms. For cloud customers, it enables on-demand resource scaling, and outsources the maintenance of computing resources. However, apart from all these benefits, it also brings a new security threat. In theory, VMs running on the same physical server are logically isolated from each other. In practice, nevertheless, malicious users can build various side channels to circumvent the logical isolation, and obtain sensitive information from co-resident VMs, ranging from the coarse-grained, e.g., workloads and web traffic rates to the fine-grained, e.g., cryptographic keys For clever attackers, even seemingly innocuous information like workload statistics can be useful [9]. For example, such data can be used to identify when the system is most vulnerable, i.e., the time to launch further attacks, such as Denial-of-Service attacks.

A straight forward solution to this novel attack is to eliminate the side channels, which has been the focus of most previous works. However, most of these methods are not suitable for immediate deployment due to the required modifications to current cloud plat-forms. In our work, we approach this problem from a completely different perspective. Before the attacker is able to extract any private information from the victim, they first need to co-locate their VMs with the target VMs. It has been shown that the attacker can achieve an efficiency rate of as high as 40%, which means 4 out of 10 attacker's VMs cancolocate with the target. This motivates us to study how to effectively minimise this value. From a cloud provider's point of view, the VM allocation policy (also known as VM placement−we use these two terms interchangeably in this paper)is the most important and direct control that can be used to influence the probability of co-location. Consequently, we aim to design a secure policy that can substantially increase the difficulty for attackers to achieve co-residence.

In our earlier work we have proposed a prototype of such a secure policy, called the previous-selected-server-first policy (PSSF). However, this prototype policy only focuses on the problem of security, and hence has obvious limitations in terms of:

1. Workload balance−Workload here refers to the VM requests. From the cloud provider's point of view, spreading VMs among the servers that have
already been switched on can help reduce the probability of servers being over-utilized, which may cause SLA (service level agreement) breaches. From the customer's perspective, it is also preferable if their VMs are distributed across the system, rather than being allocated together on the same server. Otherwise, the failure of one server will impact all the VMs of a user.

2. Power consumption−It has been estimated that the power consumption of an average datacentre is as much as 25,000 households and it is expected to double every 5 years. Therefore, managing the servers in an energy efficient way is crucial for cloud providers in order to reduce the power consumption and hence the overall cost. This has also been the focus of many previous works.

In this paper, we take all three aspects of security, workload balance and power consumption into consideration to make PSSF more applicable to existing commercial cloud platforms. Since these three objectives are conflicting to some extent, we improve our earlier policy by applying multi-objective optimisation techniques. In addition, we have implemented PSSF on the simulation environment CloudSim as well as on the real cloud platform OpenStack and performed large scale experiments that involve hundreds of servers and thou-sands of VMs, to demonstrate that it meets the requirements of all three criteria.

Specifically, our contributions include: (1) we define secure metrics that measure the safety of a VM allocation policy, in terms of its ability to defend against co-resident attacks; (2) we model these metrics under three basic but commonly used VM allocation policies, and conduct extensive experiments on the widely used simulation plat-form CloudSim to validate the models; (3) we propose a new secure policy, which not only significantly decreases the probability of attackers co-locating with their targets, but also satisfies the constraints in workload balance and power consumption; and (4) we implement and verify the effectiveness of our new policy using the popular open-source cloud software OpenStack as well as on CloudSim.

4. Issues on VM Allocation Policies

The Virtual Machine allocation policy is used to place the Virtual Machines in the physical server. The malicious user co-locates their VM to the target VM. The security, workload balance and power consumption parameters are considered in the Virtual Machine placement process. Secure metrics are defined to measure the safety of the VM allocation policy. The Balanced VM Allocation Policy is build to assign VMs in the physical servers. The Previous Selected Server First (PSSF) policy is used with security metrics. Least VM allocation policy, Most VM allocation policy and Random allocation policy are used with the workload balance parameter. The data centers

are connected to the Virtual Machines with in the same environment. The following issues are discovered from the current virtual macine allocation policies against co-residential attacks.

- The system supports centralized allocation policy only
- Live VM migration is not protected
- Multiple data center management is not supported
- The system state information is required for the scheduling process

5. Distributed Scheduling and Virtual Machine Management Framework

The virtual machine placement operations are performed with centralized and distributed manner. The virtual machines are placed with workload and energy levels. Data center selection process is used to detect the suitable data center for the workloads. The system is divided into six major modules. They are Physical server deployment, Data center management, Workload controller, Security analyzer, Centralized VM placement and Distributed VM placement.

The physical servers and virtual machines are configured I the deployment process. The data center management is build to organize the data centers and shared data items. The workload controller is used to collect workloads from the users. The security analyzer is build to estimate the security metrics for the VMs. The centralized VM placement is employed to control co-resident attacks. Live VM migration and multiple data center based allocation are performed under the distributed VM placement model.

5.1. Physical Server Deployment

The physical server deployment is used to setup the cloud with shared resources. The physical servers and configuration levels are collected from the cloud provider. The virtual machine configurations are assigned with provider choice. The physical server and virtual machine association levels are updated under the deployment process.

5.2. Data Center Management

The data centers and its contents are maintained under the data center management. Data center storage and usage levels are monitored in intervals. Shared data and its request frequency are maintained in the data center**.** Virtual machines and data center communication are controlled with security levels.

5.3. Workload Controller

The workload controller monitors the workload execution process. The workloads are collected from the cloud users. The workloads and data association are verified by the controller. The workload status is monitored and updated by the controller.

5.4. Security Analyzer

The security analyzer is used to estimate the security levels for the virtual machines. The secure metrics are used to estimate the security levels. The workload balance parameter is considered in the secure metrics. The power consumption levels are estimated in the secure metric estimation process.

5.5. Centralized VM Placement

The centralized VM placement is carried out with the balanced VM allocation policy. The Previous Selected Server First (PSSF) algorithm is used for the VM placement process. Single data center is used to provide the data values. The secure metrics are used in the VM placement process.

5.6. Distributed VM Placement

The secure metrics estimation and initial VM placement operation are tuned for the distributed scheduling model. The Distributed VM placement algorithm is used to allocate the virtual machines in distributed manner. The Data center allocation algorithm is used to select data centers for the virtual machines. The live VM migration operations are secured with Attack resistant VM migration algorithm.

6. Conclusion

Cloud computing environment provides IT resources to the users based on their demand. The Co-resident attacks are raised by the co-located malicious users with side channels. The Previous Selected Server First (PSSF) policy is applied for the VM placement with security. The attack resistant VM placement is build with centralized and distributed scheduling, Live VM migration and Multiple data center management. The Virtual Machine placement operations are carried out with side channel attack control mechanism. The VM placement policies are improved to manage centralized and distributed placement models. The co-location control model is tuned to handle the VM migration tasks. Data center communications are also protected in the allocation scheme.

References

[1] Hao Wu, Shangping Ren, Gabriele Garzoglio, Steven Timm, Gerard Bernabeu,Keith Chadwick and Seo-Young Noh, “A Reference Model for Virtual Machine Launching Overhead”, IEEE Transactions On Cloud Computing, July-September 2016.

[2] Keller, E., Szefer, J., Rexford, J., Lee, R.B.: Eliminating the hypervisor attack surface for a more secure cloud. In: Proceedings of ACM Conference on Computer and Communications, Security (CCS’11) (2011)

[3] McCune, J.M., Li, Y., Qu, N., Zhou, Z., Datta, A., Gligor, V., Perrig, A.: TrustVisor: efficient TCB reduction and attestation. In: Proceedings of

2010 IEEE Symposium on Security and Privacy, Oakland (2010)
[4] Varadarajan, V., Kooburat, T., Farley, B., Ristenpart, T., Swift, M.M.: Resource-freeing attacks: improve your cloud performance. In: Proceedings of 2012 ACM Conference on Computer and Communications Security, Raleigh (2012)
[5] Zhang, Y., Juels, A., Oprea, A., Reiter, M.K.: HomeAlone: Co- Residency Detection in the Cloud via Side-Channel Analysis. In: Proceedings of 2011 IEEE Symposium on Security and Privacy, Berkeley (2011)
[5] Zhang, Y., Juels, A., Reiter,M.K., Reiter, M., Ristenpart, T.: Cross- VMside channels and their use to extract private keys. In: Proceedings of 2012 ACM Conference on Computer and Communications Security, Raleigh (2012)
[7] Adam Bates, Benjamin Mood, Joe Pletcher, Hannah Pruse, Masoud Valafar and Kevin Butler, "On detecting co-resident cloud instances using network flow watermarking techniques", Springer-Verlag Berlin Heidelberg 2013
[8] Bowers, K.D., van Dijk, M., Juels, A., Oprea, A., Rivest R.L.: How to tell if your cloud files are vulnerable to drive crashes. In: CCS '11: Proceedings of 18th ACM Conference on Computer and Communications Security, Chicago, 2011.
[9] Eun Kyung Lee, Hariharasudhan Viswanathan and Dario Pompili, "Proactive Thermal-aware Resource Management in Virtualized HPC Cloud Datacenters", IEEE Transactions on Cloud Computing, June 2017.
[10] Butt, S., Lagar-Cavilla, H.A., Srivastava, A., Ganapathy V.: Selfservice cloud computing. In: Proceedings of 2012 ACM Conference on Computer and Communications Security, Raleigh (2012)