Journal of Information Security and Applications 000 (2020) 102397

Contents lists available at ScienceDirect

Journal of Information Security and Applications

journal homepage: www.elsevier.com/locate/jisa

A signaling game approach to mitigate co-resident attacks in an IaaS cloud environment

M.G.M. Mehedi Hasan[a,*], Mohammad Ashiqur Rahman[b,*]

[a] Department of Computer Science, Tennessee Tech University, Cookeville, USA
[b] Department of Electrical and Computer Engineering, Florida International University, Miami, USA

ARTICLE INFO

Article history:
Available online xxx

Keywords:
Co-resident attacks
Collaborative attacks
Signaling game
Nash equilibrium
Cloud security

ABSTRACT

Cloud service providers (CSPs) offer a variety of services that are opening the door to the infinite possibilities of cloud computing. Despite numerous benefits offered by the CSPs, there are, however, some security issues that may dissuade users. In cloud computing, different virtual machines (VMs) often share the same physical resources, which are known as co-resident VMs. The shared physical resources pose a significant threat to the users as resources may belong to competing organizations as well as unknown attackers. From the perspective of a cloud user, there is no guarantee whether the co-resident VMs are trustworthy. The shared resources make privacy and perfect isolation implausible, which paves the way for co-resident attacks in which a VM attacks another co-resident VM through a covert side channel that can be used to extract another user's secret information or launch denial of service attacks. The attack campaign becomes more damaging when multiple co-resident VMs collaborate. In this paper, we analyze the co-resident attacks and corresponding defense strategies, with respect to benign and malicious VMs and the defender, i.e., the VM monitor (VMM), using a signaling game model. The solutions to the game provide optimal defense strategies for the VMM with respect to the expected number of malicious VMs in collaboration. We evaluate the game results through simulations on various synthetic attack scenarios. The results show that the defender can effectively resist co-resident attacks by distinguishing the benign and malicious VMs.

1. Introduction

Cloud computing opens the door to co-resident, cross-side channel attacks between virtual machines (VMs) in a public cloud when the VMs share the same hypervisor, CPU, memory, storage, and network devices. Some of the resources can be partitioned (e.g., CPU cycles, memory capacity, and I/O bandwidth). However, VMs also share resources that cannot be well partitioned, such as last-level cache (LLC), memory bandwidth, and IO buffers. The shared resources can be exploited by attackers when launching cross-side channel attacks to extract other users' secret information or when launching a denial of service (DoS) attack.

The number of people drawn to use cloud computing services is increasing rapidly because of the varieties of services offered by the cloud service providers (CSPs) coupled with the flexibility of use and cost effectiveness. CSPs provide cloud services in different forms. Infrastructure as a service (IaaS) is one of them, which offers each cloud service user a VM that emulates a physical machine, thereby providing the user the greater flexibility an actual machine would yield. Several VMs sharing the same physical resources, like CPU, memory, and storage devices, are called co-resident VMs, which can lead to new kinds of attacks known as co-resident attacks. In these attacks, malicious users build various types of side channels [1,2] between their VMs and the target VM on the same server. These side channels are used for extracting sensitive information from the victim. At the heart of these attacks is the last level cache (LLC), which is also known as the L3 cache. The L3 cache is shared between all the residing VMs, which opens the door for the attacker to launch cache-timing attacks. Ristenpart et al. were the first to discover this kind of attack [1] that exploited cache timing. They discussed how the concept of covert channel could be extended to launch attacks in clouds. They used the idea of Bernstein [3] show that it is possible to extricate AES key using cache-timing attacks. Hence, various types of secret information including cryptographic keys can be leaked to malicious users through these side channels. Since co-resident attacks are carried out using covert channels and they leave few traces in the system logs, it is harder to detect them.

* Corresponding authors.
E-mail addresses: mmehediha42@students.tntech.edu (M.G.M. Mehedi Hasan), marahman@fiu.edu (M.A. Rahman).

https://doi.org/10.1016/j.jisa.2019.102397
2214-2126/

There are several techniques that a malicious VM uses to launch co-resident attacks. One such technique is the PRIME+PROBE technique, which is based on cache-timing [3–5]. In this case, an attacker shares the physical resources with its target victim through a VM. The attacker VM observes the cache activity of his victim VM by first priming the cache memory of the victim VM, then remaining in *busy-wait* state for a certain amount of time. During this time, the attacker waits for the victim to use the cache. After that, the attacker VM primes the cache again and if the attacker's prime results in a cache *hit*, that particular cache line was not used by the victim process. However, if the prime results in a cache *miss*, the cache line was used by the victim process, which evicted the attacker's data, thus resulting in more time to acquire the data. This cache-timing of loading the data reveals what kind of activity is executing on the victim VM. The pattern of activities makes it possible to extricate cryptographic keys. The detail technique of this attack type is discussed in the appendix of the technical report [6]. There is another co-resident attack technique known as the FLUSH+RELOAD technique [2,7,8], which requires the attacker to reside in the same core. The attacker first flushes the cache of the victim and then remains in *busy-wait* state for a certain amount of time for the victim to run its process. Now, if the attacker reloads his data and observes the timing, he can find that some data result in longer loading times. The attacker can perceive that this cache line was used by the victim process, which can be leveraged to extract the secret keys.

Co-resident attacks are normally carried out in two steps. In the first step, the attacker VM attempts to co-reside with the target VM. After achieving the co-residence, the attacker VM launches the side channel attacks as a second step. Since the co-resident attacks make the benign VMs suffer, it is important for the VM monitor (VMM) to act against them. However, the VMM should take its steps carefully so that a benign VM does not suffer and a malicious VM does not go unpunished. The main impediment that the CSP faces in this case is that the malicious VMs are also its legitimate clients. If proper caution is not taken while taking defensive action, a benign VM might get punished through a wrong decision. Therefore, it is crucial to identify potential attackers with a certain level of confidence. An attacker can strive to deploy several co-resident VMs forming a collaborative attack team to carry out the attack process. This coordinated attack is highly appealing for an attacker as the use of multiple attacking VMs, instead of one single VM, increases the chance of realizing the attack goal.

Contributions. The detection of the attacker VMs and the application of appropriate countermeasures should be done with proper discretion. We address this need in this research by modeling the co-resident attack and defense problem as a game, which we named as the co-resident attack mitigation and prevention (CAMP) game. The solution to the game allows the VMM to make a smart application of defense actions by distinguishing malicious VMs from benign VMs. We model the CAMP game as a signaling game [9]. Since the signaling game is suitable for distinguishing between different types of senders (i.e., the benign and malicious VMs) with respect to the defender (i.e., the VMM), using a belief model, it will be a perfect mechanism for defending against co-resident attacks. The proposed game assumes an expected capability for the attacker with respect to the number of VMs employed for launching the co-resident attack. The defense mechanism includes a set of security measures among which the VMM selects its strategy against the attacker VMs. In this work, we consider a cloud environment where the same VM images provide the same services i. e., we consider "standard cloud servers" instead of "clustered cloud servers."

According to this game framework, we analyze the interactions between the VMs and the VMM and obtain both the pooling and separating equilibria [10]. We evaluate the equilibrium strategies, especially with respect to the defender, by developing a simulation program and running synthetic co-resident attack scenarios. Our game model defends against any number of attacker VMs, where an attacker can deploy a single VM or multiple (collaborative) VMs. The results show that the optimal defense strategies provided by the CAMP game can foil the attacker's effort by successfully detecting the malicious VMs. A preliminary outcome of this work, especially when the attacker deploys only a single VM to carry out the attack, was published in [11].

Organization. The rest of this paper is organized as follows. In Section 2, related work on co-resident attacks is presented. Section 3 presents the strategy of our game model. We present the game model in Section 4. We analyze the game result in Section 5. The following section presents the evaluation of the game results. We further explain several intriguing points on this research in Section 7. We conclude the paper in Section 8.

2. Related work

Co-resident attacks have been studied in different research works. Han et al. explored how malicious users aim to co-locate their virtual machines (VMs) with the target VMs on the same physical server to extract private information from the victim using side channels [12]. The authors proposed how the use of game theory can help reduce these attacks through efficient VM placement, which reduces the chance of co-residency of the attacker VM with the victim VM [13]. Jensen et el. talked about malware injection attacks in [14]. In this kind of attack, a malicious attacker attempts to inject malicious services or virtual machines into the cloud. If the attacker can successfully launch malicious services, the CSP can face serious consequences. Zhang et al. showed how side channel attacks can be done by placing a malicious VM on a target cloud server [2]. If the attackers become successful, they can extract the private keys of the target VM.

Singh et al. focused on a denial of service (DoS)-based attack, where a co-resident VM can congest the network channel shared among other co-resident VMs in the cloud [15]. Liu et al. discussed how Intel's Cache Allocation Technology (CAT) can be utilized to prevent co-resident attacks by smartly managing Class of Services (CoS) [16]. Shi et al. designed a dynamic page coloring solution to limit cache side channel attacks [17]. Vattikonda et al. [18] and Wu et al. [19] worked on how the high resolution clocks, that many side channels rely on, can be modified or removed. Jin et al. [20] and Szefer et al. [21] worked on redesigning the architecture of cloud computing systems. Zhang et al. proposed how the side channel can be made noisy to defeat the attacker's effort to extract information [22]. Luo et al. pointed out how the lack of security border and isolation can allow the possibility of information leakage [23]. They proposed a divide and conquer strategy, where cloud computing virtualization problems are classified and analyzed, respectively. Based on this classification, they provided the corresponding response measures.

Manshaei et al. did an extensive survey on how game theory can be utilized to meet the network security issues [24]. That survey covered various research papers dealing with a variety of security concerns. Han et al. applied game theory in virtual machine allocation policy to prevent co-residence in the cloud [12]. Maghrabi et al. used game theory to assess the risk involved in moving critical assets of an IT system to a public cloud [25]. Qiu et al. worked on secure virtual machine deployment strategy to reduce co-residency in the cloud [26]. Wang et al. added reputation deriving from the social cloud as part of the utility and proposed the interaction between two rational parties in the social cloud as a game where two parties receive their opponent's trust or reputation from the social cloud [27]. They showed that every party has the motivation to cooperate and increase their