A Defending Technology Against Co-Resident Attack Considering Early Warning Mechanism and Disguise Component

Li Deng[1], Ailing Deng[1], Yuxi Peng[2], Yanping Xiang[1], Jianwei Xiang[3,*]

[1]University of Electronic Science and Technology of China, Chengdu, Sichuan, China
[2]Southwest Jiaotong University, Chengdu, Sichuan, China
[3]Hunan university of technology, Hunan, China

Abstract

With the development of cloud services, cloud servers must provide a safe and reliable cloud environment. To defend co-resident attack launched by malicious cloud users who co-resident with normal users on the same physical server, based on FFP voting, we propose a probabilistic model for evaluating the failure probability of an N-Version service program with disguise components and early warning agents in this paper. Under the condition of defense resource constraints, the failure prediction of the NVP service program is used as the basis to select the optimal deployment strategy of NVP, which is oriented to minimize the failure probability.

Keywords

early warning mechanism; disguise component; cloud environment; N-version programming

1. Introduction

In complex systems, especially in safety-critical systems, any software failure may bring catastrophic consequences. Intensive and thorough software testing is expensive and can not eliminate all software failures. Therefore, a more economical method is needed to improve the quality of complex systems. Software fault tolerance is such a method. The concept of NVP (N-version programming) was originally proposed by Elmendorf [1]. Specifically, different service program versions are developed to respond to a request at the same time, and the final result rest with a specific voting method based on all the outputs.

As a mature I.T. paradigm, cloud computing is widely used because of its high flexibility, scalability and high-cost performance. To meet the high-reliability requirements of key service requests, cloud service providers use redundant resources in the cloud environment to realize a variety of fault-tolerant technologies, such as NVP mentioned above. As proved by recent works, the cloud platform can help to realize NVP[2-6]

The foundation of cloud computing is to achieve high resource utilization through sharing: suppliers jointly host multiple VMS on a single hardware platform. However, virtual resources are mapped to shared physical resources, resulting in the possibility of interference between jointly hosted virtual machines. The services is vulnerable since an attacker can co-locate its V.M.s with a target VMs on a server and carry out a side-channel attack to steal or destroy the user's sensitive information.

In this paper, based on FFP voting, we propose a probabilistic model of evaluating the failure probability of an NVP service with disguise components(DC) and early warning agents(EWA). Furthermore, under the condition of defense resource constraints, the failure prediction of the program is used as the basis to select the optimal deployment strategy of NVP, which is oriented to maximize the success probability of NVP service. This is the first time to consider adding an early warning component and a disguise component simultaneously in the NVP system. And this strategy can effectively help NVP service components to resist co-resident attacks in cloud environment.

ICCEIC2022@3rd International Conference on Computer Engineering and Intelligent Control

EMAIL: 202021081206@std.uestc.edu.cn, (Li Deng) 534615669@qq.com (Ailing Deng), 413820155@qq.com (Yuxi Peng), yanping_xiang@163.com (Yanping Xiang), *xjwlfylfylpp@126.com (Jianwei Xiang)

© 2022 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

CEUR-WS.org/Vol-3344/paper29.pdf

The rest of the paper is organized as follows: Section 2 presents some relevant works. Section 3 presents the probabilistic model for assessing the failure probability of the service with DCs and EWAs. In section 4, we solve the optimization problem of finding the optimal amount of SCs, DCs and EMAs to minimize loss costs. Section 5 concludes our results and discusses our future work.

2. Related work

To protect the cloud environment from being destroyed by co-resident attacks, tremendous methods have been proposed in the literature. For instance, restricting or eliminating side channel structures. Most side channels utilize LLCs shared by different virtual machines. A simple way to eliminate LLC side sharing is to prevent it. For example, modify the hardware to divide the LLC into multiple regions of different VMs[7]. Those methods require a lot of hardware or software modifications, making it difficult for cloud service providers to adopt.

Several researchers have also studied developing a secure virtual machine allocation policy. When cloud providers allocate VMs, reducing co-residence between users can also mitigate co-resident attacks. Han [8] proposed PSSF to mitigate the co-occupancy attack. PSSF denotes the current tenant selecting the physical machine previously used to minimize the co-residence probability. This method also considers the problems of load balancing and power consumption and adopts the technique of limiting physical machines and fixed groups for users. Azar[9] proposed an anti-co-resident attack algorithm to limit the propagation of VM by maintaining a fixed number of physical machines. Similarly, Qiu[10] also proposed a virtual machine distribution strategy of "diffusion before concentration" to resist co-resident attacks.

Data partitioning technology can effectively suppress the risk of unaccredited access, because information can be stolen merely when the attacker can access the data blocks of all partitions[11] if the information is helpful only in terms of its integrity.

Replication and cancellation technique were studied in [12] and [13]. They create many task replicas to shorten the expected completion time of the task and increase data reliability. An early warning mechanism involves decisions, associated policies, and procedures designed to predict and mitigate network attacks based on specific network threat intelligence. Several individuals have researched the detection of co-resident attacks, such as classifying users through semi-supervised learning to identify possible malicious users and provide other users with early warnings.

3. Failure probability evaluation

3.1. Introduction of D.C. and EMA

Reference[14] proves that distributed disguise components(DC) can reduce the probability of essential components being attacked, so as to improve the reliability of multi-component systems. In order to reduce the probability of service corruption provided by NVP redundant components, this paper considers using defense processes similar to honeypots as a disguise component mechanism to deploy security resources, which can distract attackers from more valuable service components on the network. The attacker cannot differentiate DC from SC and attempts to establish side channels for any co-existing DC. Suppose the attacker's virtual machine co-resides with SC and the disguise component. In that case, AVM has a 0.5 probability of attacking the disguise component to reduce the probability that SC will be attacked and affect the voting results.

To protect the service from being corrupted, the cloud providers meanwhile distributes e EWAs including attack monitoring system and detection software in servers which having SCs. AVMs cannot distinguish them and will seek to establish side channels, which the co-resident EWA can detect using side-channel or cache usage and obtain facts or details about the attack by p probability. The EWA can contacts with other servers and provides the attack's information if the attack is successfully detected, which prevents all the AVM from performing an attack. Consequently, as long as one attack from an AVM is identified, there will be no SC damage from co-resident attacks.

3.2. First-past-the-post voting mechanism

Most crucial businesses often utilize a specific redundancy technology to achieve high reliability. In particular, in the cloud system running NVP services, various service components(SC) are located on physical servers and perform one task simultaneously. The final output is provided to service users by voting on the outcomes of these different SCs. There are numerous voting rules, including threshold voting, which selects output with more votes than a preset threshold value. In some particular cases, the majority voting system has a majority output (i.e., a threshold of 50% is selected).

The first-past-the-post voting mechanism chooses the winner who gets the most votes, regardless of whether its number exceeds the threshold of 50%. When the amount of damaged SCs is bigger than the amount of undamaged SCs that provide correct output, the attacker have the victory, representing the destruction of the entire NVP service.

This voting mechanism is more real-time than the majority voting system because of the existing time competition. It also emphasizes the competition between the SC's service task execution process and the attacker's virtual machine.

3.3. Evaluation of damage probability

3.3.1. Failure probability

The cloud service provider assigns n (1≤n≤s) service components in n servers. If the total number of the servers is o, a fixed subset of size n holds SC in the o servers.

According to the formulas proposed in [15] ,there are o^b ways to distribute b AVMs on o servers. Based on the principle of tolerance and exclusion, we can get that the number of allocation methods for the co-resident of AVM and SC is $\sum_{i=0}^{t}(-1)^{t-i}\binom{t}{i}(o-n+i)^b$, where AVMs and SCs co-resident in a fixed subset of t servers。

The total probability of v SCs being destroyed by attackers is (k present the success probability of AVM damage to co-resident SC):

$$p(o,n,b,v)=\sum_{t=v}^{\min(b,n)} g(o,n,b,t)\binom{t}{v}k^v(1-k)^{t-v} \tag{1}$$

Let c denotes the probability that each undamaged SC provides correct output, then given that v is the number of damaged SC, the conditional probability of damage to the whole service component is:

$$w(n,v)=\begin{cases}\sum_{j=0}^{v-1}\binom{n-v}{j}c^j(1-c)^{n-v-j}, & \text{if } v\le n/2\\ 1, & \text{if } v>n/2\end{cases} \tag{2}$$

Therefore, the damage probability of an NVP service is:

$$f(o,n,b)=\sum_{v=2}^{\min(n,b)} p(o,n,b,v)w(n,v) \tag{3}$$

3.3.2. Probability of NVP service failure

According to the formulas proposed in [15], the probability of EMA co-residing with AVM when EMA is available on e servers with SC is

$$\binom{e}{\theta}\binom{n-e}{t-\theta}\binom{n}{t}^{-1}$$

Suppose EMA can detect co-residence with a probability of z, the probability of at least one AVM being detected when SC co-resides with AVM in x servers is:

$$h(n,e,t) = \frac{\sum_{\theta=1}^{\min(t,e)} \binom{e}{\theta}\binom{n-e}{t-\theta}[1-(1-p)^{\theta}]}{\binom{n}{t}} \tag{4}$$

The conditional probability of AVM successfully damaging v SCs when SC co-resident with AVM in t servers is

$$[1 - h(n,e,t)] \binom{t}{v} k^{v}(1-k)^{t-v}$$

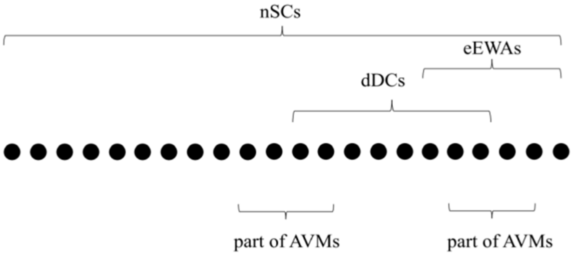

Figure 1 Co-residence of EWAs, DCs and AVMs

From Figure 1, there are b AVMs co-residing with SCs, from which a portion co-resides with EWAs and another co-resides with DCs. A section co-reside with both DCs and EWAs, while the remainder does not co-reside with either DCs or EWAs. The probability of a DC out of d co-residing with AVM is:

$$\binom{d}{a}\binom{n-d}{t-d}\binom{n}{t}^{-1}$$

The attacker cannot distinguish between DC and SC and hence will attempt to create side channels for any co-resident DC. If the attacker's virtual machine co-resides with SC and the DC, AVM has a 0.5 probability of attacking the disguise component. At the same time, EWA can also identify the attack.

Consequently, the conditional probability of AVM successfully damaging y SCs when SC co-resides with AVM in x servers is:

$$[1 - h(n,e,t)] \sum_{\delta=0}^{v} \binom{a}{\delta} \left(\frac{k}{2}\right)^{\delta} \left(1 - \frac{k}{2}\right)^{a-\delta} \binom{t-a}{v-\delta} k^{v-\delta}(1-k)^{t-a-v-\delta} \tag{5}$$

Furthermore, the probability of y SCs being damaged is:

$$p(o,n,e,b,v,d) = \sum_{t=v}^{\min(n,b)} \sum_{\delta=0}^{v} [1 - h(n,e,t)] \binom{a}{\delta} \left(\frac{k}{2}\right)^{\delta} (1 - \frac{k}{2})^{a-\delta} \binom{t-a}{v-\delta} k^{v-\delta}(1-u)^{t-a-v-\delta} \tag{6}$$

Hence, the total failure probability is:

$$\text{f(o,n,e,b,d)} = \sum_{v=2}^{\min(n,b)} p(o,n,e,b,v,d)w(n,v) \tag{7}$$

3.3.3. Effect of model parameters on PSC

With other parameters unchanged, the probability of co-residence raises when the number of attacks b raises, leading to an raise in the failure probability f. Similarly, when the early warning component e increases, so does the probability of detecting co-resident attacks, which decreases the failure probability f. Additionally, the probability of an attacker corrupting a service component reduces as the number of disguise components d increases, also leading to a decrease in the failure probability f.

4. Formulation of optimal strategy

4.1. Optimization of the cloud service

This section provides a solution for the optimal number of services components, early warning agent and disguise components under resource constraints which oriented to minimize the failure probability of NVP services according to the method mentioned in[15]. Assuming that the limited defense resource is R_U.

4.1.1. The number of AVMs is certain

$C_U(n, e, d)$represents the overhead of creating a virtual machine running service component, EWA and DC, assigned to different users. Moreover, $C_U(n, e, d)$ is ostensibly an increasing function of the numbers n, e and d. We can assume that $C_U(n, e, d) = c_n * n + c_d * d + c_e * e$, where c_n, c_d and $c_e$respectively represents the overhead of working on a single SC, DC and EWA. After normalization, we can express it as:

$$C_U(n, e, d)/c_n = n + c_d/c_n * d + c_e/c_n * e \tag{8}$$

To minimize $f(o, n, e, b, d)$, the optimization problem can be formulated as:

$$n^*, e^*, d^* = \arg_{n,e,d}\min f(o, n, e, b, d),$$

$$\text{s.t.}\ \ n + c_d/c_n * d + c_e/c_n * e \leq R_U/c_n; \tag{9}$$

We can therefore use the brute force enumeration to solve this optimization problem since n, e and d are integers.

4.1.2. The number of AVMs is uncertain

However, in most cases, the number of AVMs m is uncertain. Assuming we can acquire the distribution information of m through historical data and expert help, $\mu(t) = \Pr(b = t)$ where $b_{\min} \leq t \leq b_{\max}$.

$$n^*, e^*, d^* = \arg_{n,e,d}\min \sum_{t_{\min}}^{t_{\max}} \mu(t)\, f(o, n, e, b, d)$$
$$\text{s.t.}\ \ n + c_d/c_n * d + c_e/c_n * e \leq R_U/c_n \tag{10}$$

4.2. Optimization considering the attacker's behavior

Let $C_A(b)$ represents the overhead of creating a virtual machine and launching attacks, $C_A(b) = b * c_b$, where c_b represents the overhead of creating a virtual machine and launching an attack. If the limited attack resource is R_A;

For an attacker, to seek the maximum attack winning probability:

$$b^* = \arg_b \max f(o, n, e, b, d).$$
$$\text{s.t.}\ \ b \leq R_A/c_b \tag{11}$$

Take attacker's behavior into consideration, the optimization problem can be rephrased as:

$$n^*, e^*, d^* = \arg_{n,e,d}\min f(o, n, e, b, d)),$$

$$\text{s.t.}\ \ n + c_d/c_n * d + c_e/c_n * e \leq R_U/c_n \tag{12}$$

5. Conclusion

In this paper, we model a fault-tolerant NVP service by implementing disguise components and early warning agents to resist co-resident attacks. We developed a model to assess the failure probability of the program considering the effects of EWA and DC. Furthermore, we study the relevance between service failure probability and parameters, containing the number of SCs, the number of DCs, the number of EWAs, the correct execution probability of undamaged SCs, the detection probability of EWAs, and the damage probability of SCs through their co-resident AVMs. Finding the best allocation strategy for both attacker and defender is crucial. Hence, we developed and solved the optimization problem by identifying the best number of SCs, EWAs, and DCs to minimize the expected loss cost of service users under limited defense resources, considering the attacker's behavior.

There are several kinds of attacks in the cloud environment. However, this paper only analyzes the threat model for co-resident attacks, one of the numerous attacks in the cloud environment. In future works, we intend to pay attention to other malicious attacks with popular characteristics, such as DDoS attacks, computer viruses, node attacks, etc. Additionally, we consider using machine learning algorithms to classify users and detect the virtual machines applied by malicious users who may launch co-resident attacks to provide early warnings to other users.

6. References

[1] Elmendorf. WR. FAULT TOLERANT PROGRAMMING.[J]. IBM Tech Disclosure Bull, 1972.

[2] Pramila S, Poonkuzhali S, Mythili S. Improvising reliability through N-version programming in cloud environment. Int J Adv Tech Eng Sci April 2015;3(1):204–8.

[3] Liu J, Yang N. Proc. of 2017 7th IEEE International Conference on Electronics Information and Emergency Communication (ICEIEC). Optimal fault tolerant service provisioning for cloud application. Macau 2017:189–94.

[4] F.Khomh, "On improving the dependability of cloud applications with fault-tolerance," Proc WICSA, Article No.2,pp.1–3, https://doi.org/10.1145/2578128. 2578228, April 2014.

[5] Wagner B, Sood A. Economics of Resilient Cloud Services. Proc 2016 IEEE Int Conf Softw Qual Reliab Secur Compan (QRS-C) 2016:368–74. https://doi.org/10.1109/ QRS-C.2016.56.

[6] C.R.White, "Cloud Computing and SBSE," In: RuheG., ZhangY. (eds) Search Based Software Engineering. SSBSE 2013. Lecture Notes in Computer Science, vol 8084. Springer, Berlin, Heidelberg, 2013.

[7] Liu. F, Ge. Q, Yarom. Y, et al. CATalyst: Defeating last-level cache side channel attacks in cloud computing[C]. Proceedings-International Symposium on High-Performance Computer Architecture. 2016.

[8] Han. Y, Chan. J, Alpcan. T, et al. Using Virtual Machine Allocation Policies to Defend against Co-Resident Attacks in Cloud Computing[J]. IEEE Transactions on Dependable and Secure Computing, 2017.

[9] Azar. Y, Kamara. S, Menache. I, et al. Co-location-resistant clouds[C]. Proceedings of the ACM Conference on Computer and Communications Security. 2014.

[10] Qiu. Y, Shen. Q, Luo. Y, et al. A secure virtual machine deployment strategy to reduce co-residency in cloud[C]. Proceedings - 16th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, 11th IEEE International Conference on Big Data Science and Engineering and 14th IEEE International Conference on Embedded Software and Systems. 2017.

[11] Shinde Y, Vishwa A. Privacy preserving using data partitioning technique for secure cloud storage. Int J Comp Appl (0975 – 8887) April 2015;116(16).

[12] Luo L, Xing L, Levitin G. Optimizing dynamic survivability and security of replicated data in cloud systems under co-residence attacks. Reliability Engineering and System Safety December 2019;192:106265.

[13] Levitin G, Xing L. Co-residence based data theft game in cloud system with virtual machine replication and cancellation. Reliability Engineering and System Safety 222, 2022: 108415.

[14] McQueen. M. A, Boyer. W. F, Flynn. M. A, et al. Time-to-compromise model for cyber risk reduction estimation[J]. Advances in Information Security, 2006.I. S. Jacobs and C. P. Bean, "Fine

particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[15] Levitin G, Xing L, Xiang YP. "Optimal early warning defense of N-version programming service against co-resident attacks in cloud system" , Reliability Engineering & System Safety, 2020.