Future Generation Computer Systems 70 (2017) 17–25

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

Optimal data partitioning in cloud computing system with random server assignment

Gregory Levitin [a,b,*], Liudong Xing [c], Yuanshun Dai [a]

[a] Collaborative Autonomic Computing Laboratory, School of Computer Science, University of Electronic Science and Technology of China, China
[b] The Israel Electric Corporation, P. O. Box 10, Haifa 31000, Israel
[c] University of Massachusetts, Dartmouth, MA 02747, USA

HIGHLIGHTS

- Co-residence attacks on data distributed among virtual machines are considered.
- Model of full co-residence coverage probability is developed for cloud systems.
- The optimal data partition policy is considered.
- The minmax partition policy problem for strategic attackers is formulated and solved.

ARTICLE INFO

Article history:
Received 19 September 2016
Received in revised form
18 December 2016
Accepted 19 December 2016
Available online 23 December 2016

Keywords:
Cloud computing
Virtual machine
Co-residence attack
Data partitioning
Optimization
Minmax

ABSTRACT

Cloud computing provides a paradigm where users can utilize various configurable IT resources in an on-demand and cost-effective manner. However, new security risks such as co-resident attacks have arisen. This paper models a situation when a user partitions and distributes sensitive data among several virtual machines to make unauthorized access to the entire data difficult in a cloud environment subject to the co-resident attacks. The attacker creates virtual machines in the same environment aiming to get access to users' data. The cloud resource management system distributes all virtual machines among servers at random. The unauthorized access to data associated with user's virtual machine is possible only if this machine co-resides in the same server with the attacker's virtual machines. It is assumed that creating a side channel and getting access to the data is a common event for all the servers in which user's and attacker's virtual machines co-reside. Based on the suggested probabilistic model, an optimal number of user's virtual machines (i.e., number of different data blocks partitioned) is obtained for a fixed or an uncertain number of attacker's virtual machines, and for the case where the attacker knows the number of user's virtual machines and responds optimally on any number of these machines. Examples demonstrate that the proposed optimal data partitioning policy can effectively mitigate effects of the co-resident attacks through minimizing user's losses.

1. Introduction

Based on the virtualization technology, cloud computing offers a paradigm that allows users to utilize diverse computing and storage resources in an on-demand and cost-effective manner [1–3]. With the virtualization, cloud providers can instantiate multiple virtual machines (VMs) on a single physical server. While VMs are designed with logic isolation from their underlying hardware and from other VMs running on the same physical server [4], new security risks have emerged to exploit the co-resident VM architecture in the cloud environment. Particularly malicious attackers, through co-locating their VMs with the target user's VMs on the same physical server, can establish side-channels to bypass the logical isolation, and thus to access user's sensitive data from the co-resident VMs [5–7]. Such attacks are referred to as co-resident attacks [8–10].

Various solutions have been proposed to defend against the co-resident attacks in the cloud computing environment. For example, in [11–15] different schemes based on limiting or eliminating the construction of side-channels were developed, which typically require modifications or redesigns to the existing

* Corresponding author at: The Israel Electric Corporation, P. O. Box 10, Haifa 31000, Israel.
E-mail addresses: levitin@iec.co.il (G. Levitin), lxing@umassd.edu (L. Xing).

http://dx.doi.org/10.1016/j.future.2016.12.025
0167-739X/

Nomenclature

Acronyms

AVM	Attacker virtual machine
FCC	Full co-residence coverage
IT	Information technology
RMS	Resource management system
UVM	User virtual machine
VM	Virtual machine
VPC	Virtual private cloud

Notation

n	Number of servers in cloud computing system
k	Number of UVMs created by cloud RMS
m	Number of AVMs created by cloud RMS
v	Probability of AVM's success in getting unauthorized access to data of UVM residing in the same server
p	Probability that any UVM co-resides with at least one AVM, i.e., FCC probability
$C_U(i)$	Overhead associated with creating i UVMs
$C_A(i)$	Overhead associated with creating i AVMs
c_U	Cost of creating single VM
D	User's damage associated with unauthorized access to sensitive data
B	Attacker's benefit associated with unauthorized access to sensitive data
L	Expected total user's losses
U	Expected attacker's utility
l	Normalized user's losses $l = L/c_U$

system architecture. In [16,17], defense mechanisms based on detecting abnormal behavior of system components (e.g., CPU, cache) were proposed to mitigate effects from the co-resident attacks. In [18], a mitigation scheme based on the Intel cache allocation technology was proposed to defend against co-resident attacks on last-level cache in cloud servers using multicore processors. In [19], a defensive mechanism called HomeAlone was introduced to explore side-channels for detecting undesired co-residency. In [20,21], another technique based on network flow watermarking was proposed for detecting malicious co-resident VMs. In [22], a defensive network-based service named virtual private cloud (VPC) was introduced to suppress threats of co-resident attacks in the Amazon Elastic computer cloud. Effectiveness of the VPC technology was further verified by studies in [23]. In [24–26], VM allocation policies were proposed to increase the attackers' difficulty of co-locating their VMs with the target user's VMs. In [27,28], a defensive approach based on the gaming theory was proposed to make the co-residence difficult, thus decreasing the occurrence probability of co-resident attacks. In [29], a two-player gaming model was implemented for defending against specific co-resident Denial of Service attacks.

While the existing works have focused on coping with side channels or VMs, in this work we propose a new solution to addressing the co-resident attacks from the original user's requests point of view based on the data partition technique. The proposed solution requires no modifications of the existing system architecture or actions from the cloud providers and determines the optimal user's data partitioning policy for given parameters of a cloud environment.

In many cases the information can be useful only in its integrity [30,31]. By dividing the information into multiple separately stored data blocks, the data partitioning technique has been applied to effectively mitigate risks of unauthorized access in traditional information systems, and distributed computing systems and databases [30,32–34]. The partition technique has also recently been extended to the cloud environment [31,35–45]. For example, in [35,37], partitions were utilized for mobile data stream applications or web applications with the goal of improving the system throughput but not the data security. In [31,39] a stripping algorithm using data partition and image analysis was proposed to secure picture data containing sensitive information in clouds. In [43] data partitions were coupled with a remote data backup algorithm to improve security and integrity of data stored on cloud servers. To the best of our knowledge, however, no existing works have addressed co-resident attacks using the data partitioning technique. This work makes original contributions by proposing a probabilistic co-residence coverage model and optimal data partitioning policy to mitigate effects of the co-resident attacks. Effectiveness of the proposed methodology is demonstrated using examples.

The remainder of the work is organized as follows: Section 2 describes system and attack models. Section 3 presents the proposed probabilistic full co-residence coverage model. Section 4 formulates the optimal data partition problem and presents examples solutions. Section 5 is dedicated to the formulation and solution to the optimal data partition problem for cases with strategic attackers. Section 6 concludes the paper and gives directions for future research.

2. System and attack model

There are n servers in a cloud computing system. To complicate an unauthorized access to sensitive information, a user divides its data into k subsets and sends k requests to the cloud system. The cloud resource management system (RMS) creates a user virtual machine (UVM) for each user request (data subset) and assigns the UVMs to servers totally at random. Any server can get any number of UVM assignments from 0 to k. The UVMs can be distributed among from 1 to $\min(n, k)$ servers.

An attacker tries to get access to user's data. The access makes sense only if the attacker obtains the data in its integrity i.e. gets access to all the UVMs. To do so the attacker submits m requests to the same cloud system. The RMS creates an attacker virtual machine (AVM) for each attacker request and also randomly distributes the AVMs among servers. If an AVM co-resides with some UVMs in the same physical server, the attacker can build a side channel to each of those UVMs and get access to its part of data with a certain probability [5].

The attacker's access to user's complete data is possible only if AVMs are assigned to all servers where UVMs reside, i.e. a full co-residence coverage (FCC) is achieved. In other words, the attacker succeeds if after the random virtual machine distribution any server containing at least one UVM contains at least one AVM. The FCC probability $p(n, k, m)$ depends on number of servers n, number of UVMs k and number of AVMs m.

We assume that all servers use the same data protection measures. In this case the event of attacker's success in building the side channel and accessing data is common for all servers where UVMs and AVMs co-reside; if an AVM succeeds to build the side channel, it happens in all servers. Therefore the probability of unauthorized access to user's data is $v \cdot p(n, k, m)$ where v is the success probability of AVM getting access to data of UVM residing in the same server. We assume also that the probability v does not depend on number of UVMs and AVMs residing in the same server.

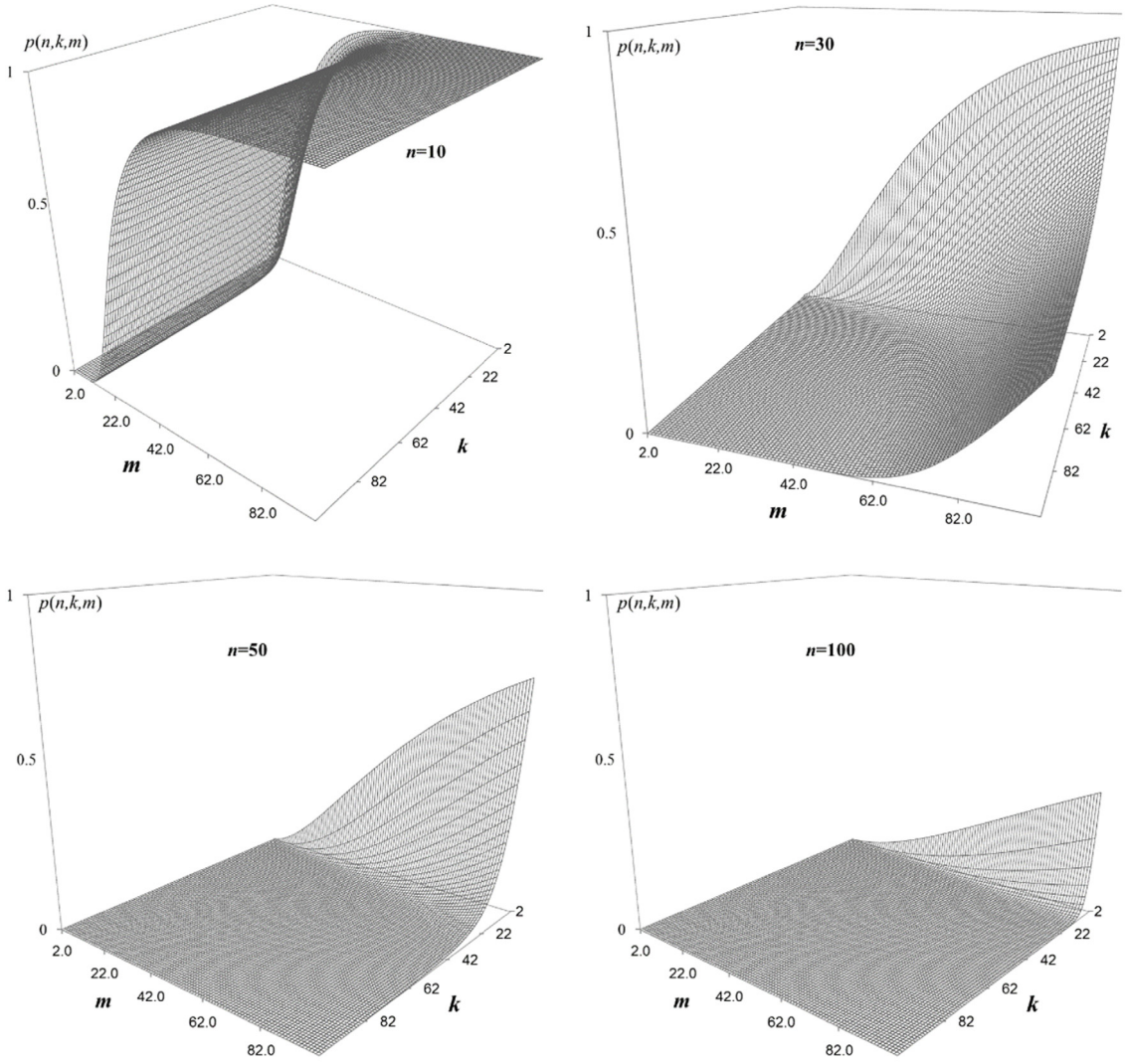


Fig. 1. Probability of FCC as a function of k and m for $n = 10$, 30, 50 and 100.

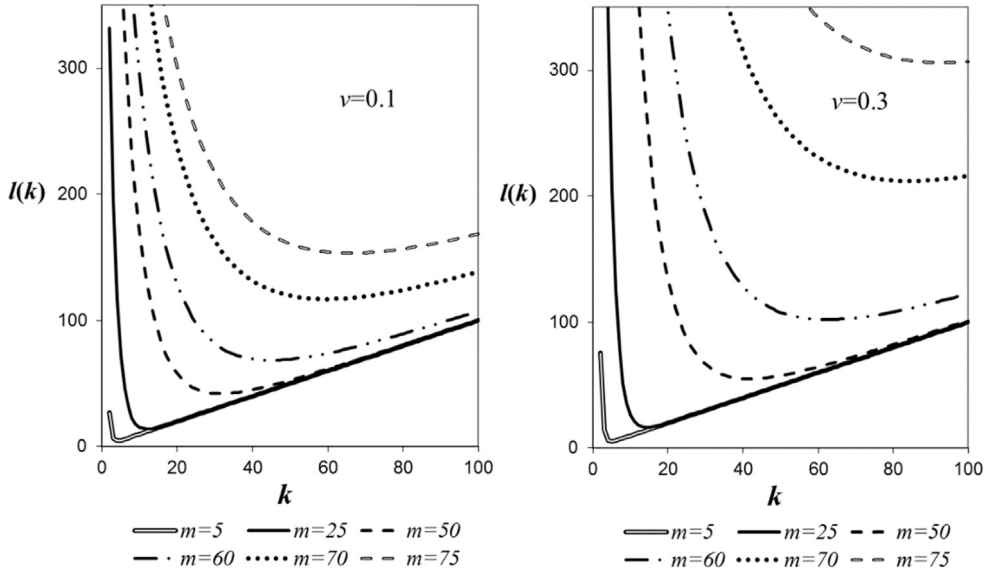


Fig. 2. Normalized user's losses as function of k for $n = 30$, $D/c_U = 10\,000$ and different values of v and m.

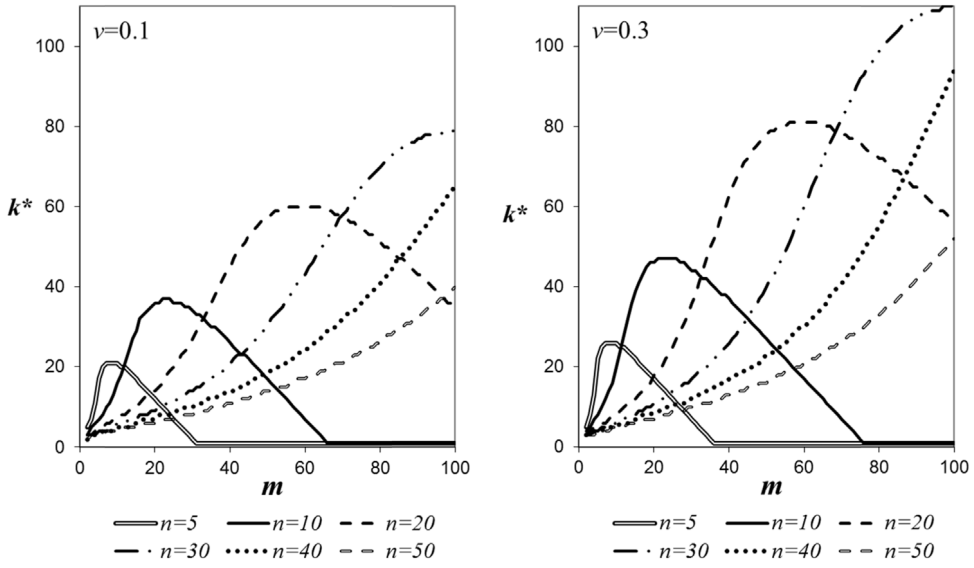


Fig. 3. Optimal value of k as function of m for $D/c_U = 10\,000$ and different values of v and n.

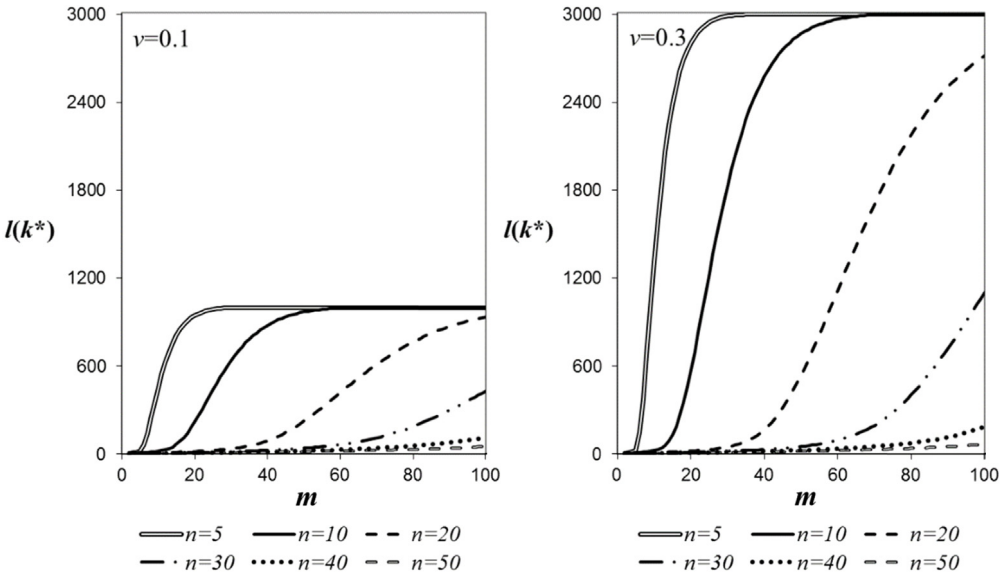


Fig. 4. Minimal normalized user's losses as function of m for $D/c_U = 10\,000$ and different values of v and n.

3. Probability of FCC

The total number of possible assignments of k UVMs among n servers is n^k. The total number of different groups of h servers among n servers is $\binom{n}{h}$. For any group of h servers the number of possible assignments of k UVMs in which i specific servers in the group host no UVMs is $(h-i)^k$. Thus, for any specific group of h servers one can obtain the number of assignment combinations in which any server gets at least one UVM using the inclusion–exclusion principle:

$$\sum_{i=0}^{h-1}(-1)^i\binom{h}{i}(h-i)^k. \tag{1}$$

The probability that exactly h servers are assigned to host k UVMs is

$$q(n,k,h) = n^{-k}\binom{n}{h}\sum_{i=0}^{h-1}(-1)^i\binom{h}{i}(h-i)^k. \tag{2}$$

Assume that fixed h out of n ($1 \le h \le \min(n,k)$) servers are assigned to host the UVMs. The total number of possible assignments of m AVMs among n servers is n^m. The total number of possible assignments in which m AVMs are distributed among no more than $n-i$ servers is $(n-i)^m$. When a fixed set of h out of n servers host UVMs, the number of possible assignments in which UVMs and AVMs do not co-reside in i specific servers hosting UVMs is $(n-i)^m$. Using the inclusion–exclusion principle one obtains the number of possible assignments in which any of h servers UVMs and AVMs co-reside as

$$\sum_{i=0}^{h}(-1)^i\binom{h}{i}(n-i)^m. \tag{3}$$

Thus, the probability that, for any specific combination of h servers hosting UVMs, any of such servers hosts also at least one of m AVMs is

$$g(n,m,h) = \begin{cases} n^{-m}\sum_{i=0}^{h}(-1)^i\binom{h}{i}(n-i)^m & \text{if } m \ge h \\ 0 & \text{if } m < h. \end{cases} \tag{4}$$

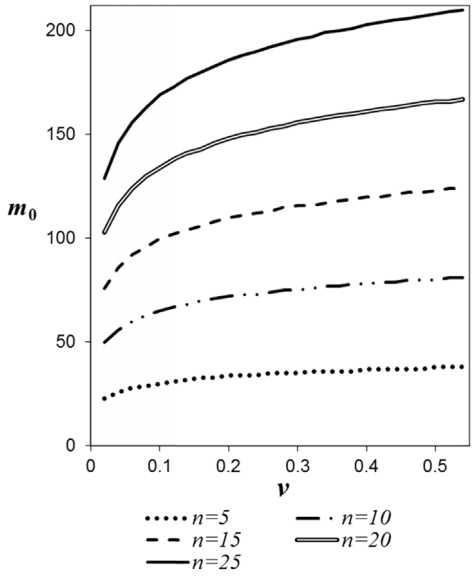


Fig. 5. Minimum number of AVM for which the data partitioning becomes non-beneficial as function of v and n.

The total probability of FCC is

$$p(n,k,m) = \sum_{h=1}^{\min(n,k,m)} q(n,k,h)g(n,m,h) = n^{-(k+m)} \sum_{h=1}^{\min(n,k,m)} \binom{n}{h} \left(\sum_{i=0}^{h-1} (-1)^i \binom{h}{i} (h-i)^k \right) \times \left(\sum_{i=0}^{h} (-1)^i \binom{h}{i} (n-i)^m \right). \quad (5)$$

Fig. 1 presents functions $p(n, k, m)$ for $n = 10$, 30, 50 and 100.

4. Optimal data partition for a fixed or uncertain number of AVMs

Creating each UVM is associated with a certain overhead for the user. The total user's overhead $C_U(k)$ is an increasing function of the total number of UVMs created k. The user can estimate the damage D associated with an unauthorized access to the sensitive data. Thus, the expected cost of losses for the user is

$$L(k) = C_U(k) + p(n,k,m) \cdot v \cdot D. \quad (6)$$

The user chooses the number of UVMs k that minimizes L:

$$k^* = \arg\min_k \left(C_U(k) + p(n,k,m) \cdot v \cdot D \right). \quad (7)$$

For the most common special case when $C_U(k) = c_U k$, where c_U is the cost of creating single UVM,

$$k^* = \arg\min_k \left(c_U \cdot k + p(n,k,m) \cdot v \cdot D \right) = \arg\min_k \left(k + p(n,k,m) \cdot v \cdot D/c_U \right). \quad (8)$$

As k is integer the optimization can be easily accomplished by brute force enumeration.

Fig. 2 presents the normalized expected user's losses $l(k) = L(k)/c_U$ as functions of the number of UVMs (k) for $n = 30$, $D/c_U = 10\,000$ and different values of v and m. It can be seen that this function has a minimum, which depends on m and v.

Figs. 3 and 4 present the number of UVMs k^*, that minimizes the expected users losses and corresponding value of the losses $l(k^*)$ as functions of number of AVMs m for $D/c_U = 10\,000$ and different values of v and n. It can be seen that $k^*(m)$ is a non-monotonic function. When the number of AVMs m is relatively small, the user can compensate the increase of m by increasing the number of UVMs k. The damage reduction caused by the increasing k exceeds the overhead. However, when the number of AVMs is large enough, the damage reduction caused by the increasing k becomes much smaller and does not exceed the additional overhead. Thus, with an increase in m the increase of k becomes non-beneficial and the optimal number of UVMs k^* decreases. Eventually, when m exceeds some value m_0, any data partitioning becomes non-beneficial and $k^* = 1$.

Fig. 5 presents the minimum number of AVMs m_0, for which data partitioning becomes non-beneficial as function of v and n. It can be seen that m_0 increases with increases in both v and n.

When the user does not know the exact number of AVMs, but has information about its distribution in the form $\mu(x) = \Pr(m = x)$ for $m_{\min} \le x \le m_{\max}$, the optimal user's policy minimizing the

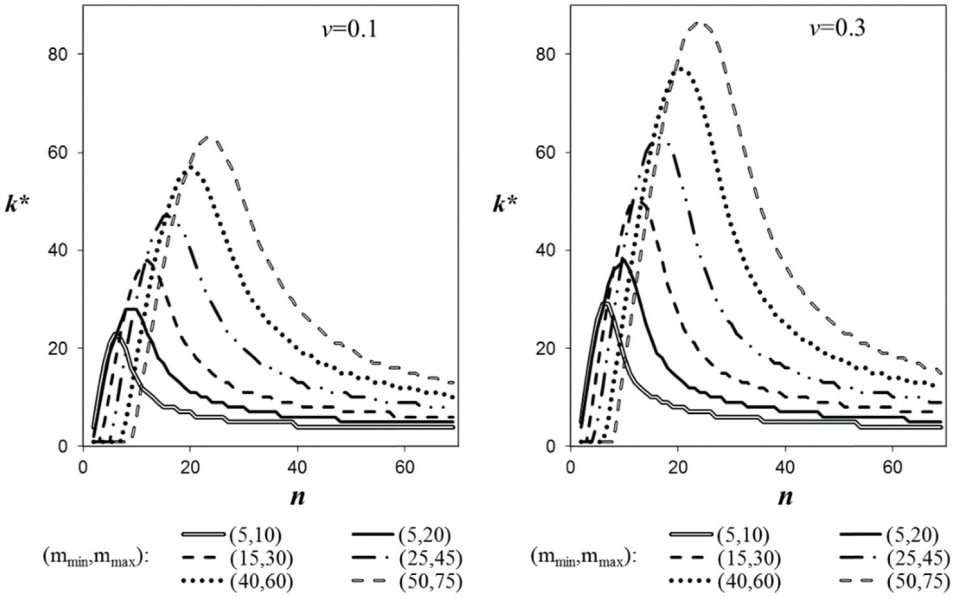


Fig. 6. Optimal value of k as function of n for $D/c_U = 10\,000$ and different ranges of m.

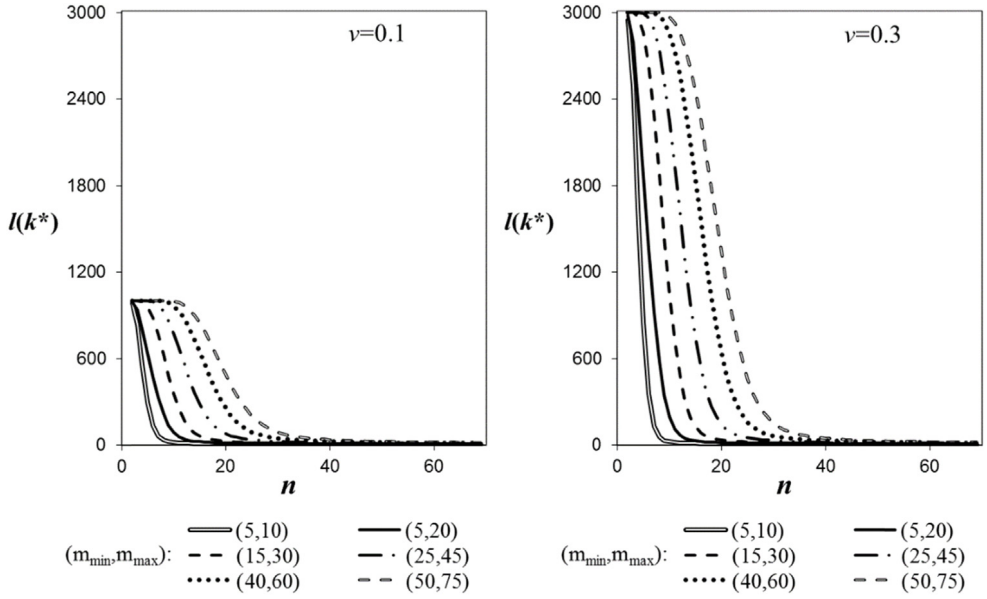


Fig. 7. Minimal normalized user's losses as function of n for $D/c_{\mathrm{U}} = 10\,000$ and different ranges of m.

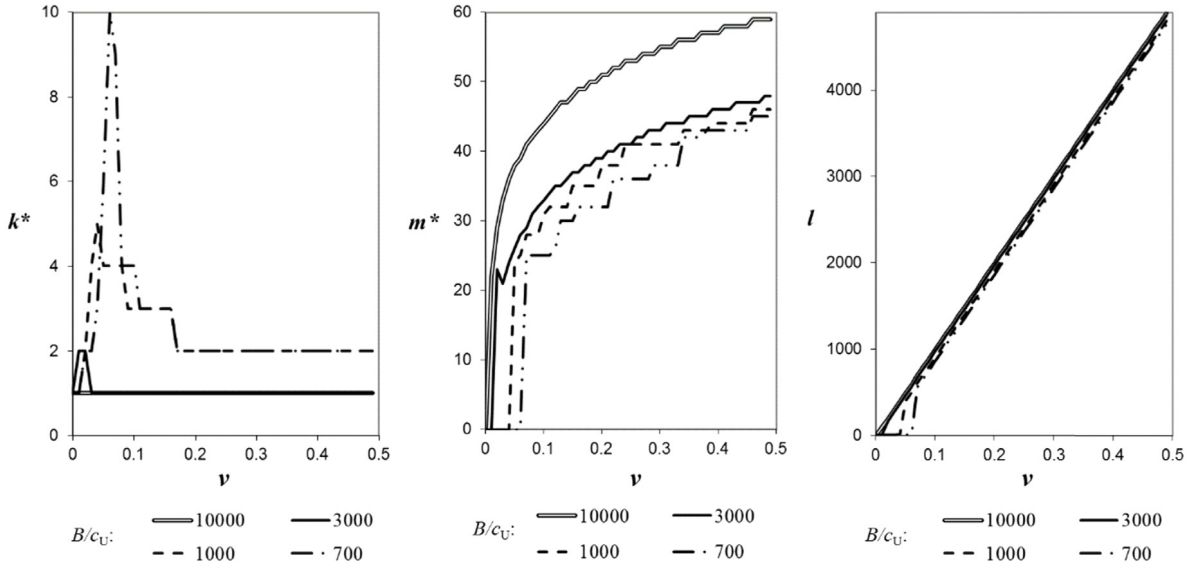


Fig. 8. Minmax solutions k^*, m^* and corresponding normalized user's losses l as functions of v and B/c_{U} for $n = 10$, $D/c_{\mathrm{U}} = 10\,000$.

expected losses is

$$k^* = \arg\min_k \left(C_U(k) + v \cdot D \sum_{x=m_{\min}}^{m_{\max}} \mu(x) p(n,k,x) \right). \tag{9}$$

Figs. 6 and 7 present the number of UVMs that minimizes the expected users losses k^* and corresponding value of the normalized losses $l(k^*)$ as functions of number of servers n, for different ranges of uniformly distributed values of m when $C_U(k) = c_{\mathrm{U}}k$, $D/c_{\mathrm{U}} = 10\,000$ and $v = 0.1, v = 0.3$. For very small number of servers n and large number of AVMs m, the probability of FCC is close to 1 and an increase in the number of UVMs cannot reduce it considerably. Thus, the data partitioning is not beneficial and $k^* = 1$. With an increase in n, the user benefits from using more UVMs and $k^*(n)$ increases. However, when n is large enough, the probability of FCC becomes negligible and an increase in k cannot reduce it considerably. Therefore, $k^*(n)$ decreases when n is large.

5. Strategic attacker

The most conservative user's policy [46] is to anticipate that the attacker guesses or chooses (based on the knowledge about the number of UVMs) the number of AVMs maximizing the attacker's utility that takes the following form

$$U(k,m) = p(n,k,m) \cdot v \cdot B - C_A(m), \tag{10}$$

where B represents the attacker's benefit associated with an unauthorized access to user's sensitive data.

Assuming that for any chosen k the attacker responds with m maximizing $U(k,m)$, the user chooses the number of UVMs by solving the following minmax problem:

$$k^* = \arg\min_k \left(C_U(k) + p(n,k,m^*(k)) \cdot v \cdot D \right) \tag{11}$$

where $m^*(k) = \arg\max_m (p(n,k,m) \cdot v \cdot B - C_A(m))$.

Notice that if $m^* = 0$, the attacker makes no attempt to get access to the user's information because the expected benefit of

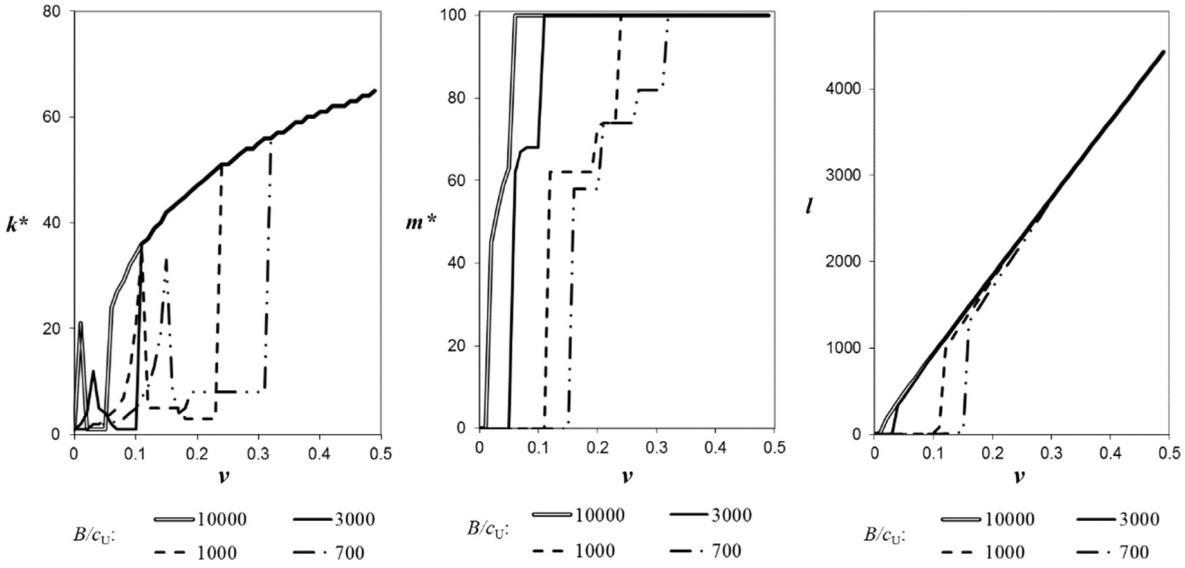


Fig. 9. Minmax solutions k^*, m^* and corresponding normalized user's losses l as functions of v and B/c_U for $n = 20$, $D/c_U = 10\,000$.

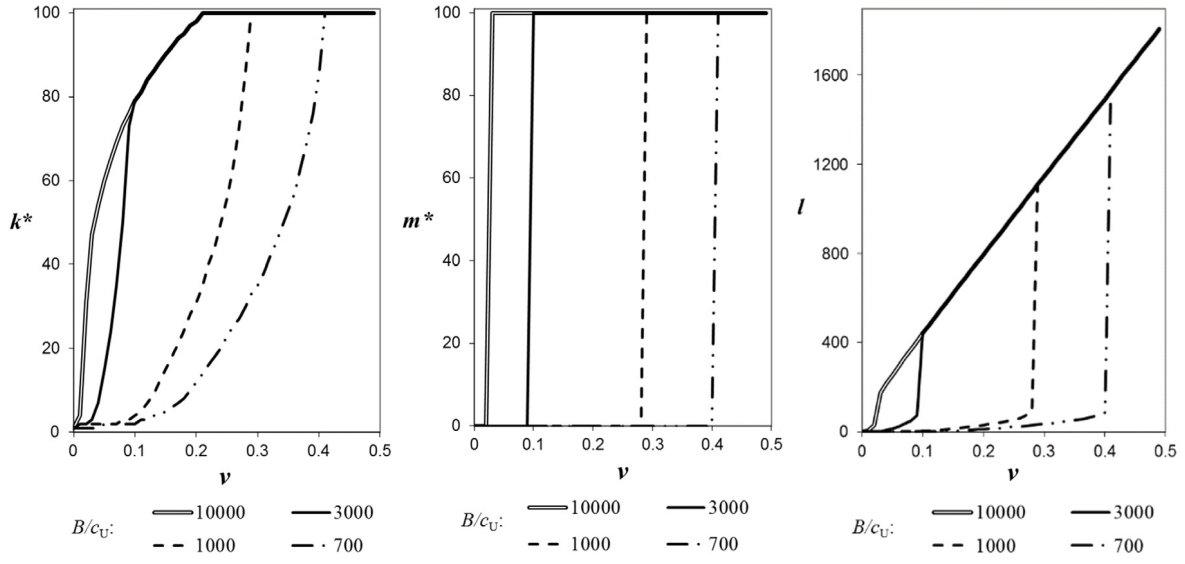


Fig. 10. Minmax solutions k^*, m^* and corresponding normalized user's losses l as functions of v and B/c_U for $n = 30$, $D/c_U = 10\,000$.

getting the information is less than the overhead associated with using the cloud system.

Figs. 8–10 present the minmax solutions k^*, $m^*(k^*)$ and the corresponding expected user's losses l assuming that $C_A(i) = C_U(i) = c_U i$ and that the number of virtual machines created for user or attacker cannot exceed 100.

It can be seen that when the probability v is very low, the attacker's chances to access the data are negligible and it prefers to avoid the attack ($m^* = 0$). To keep this situation the number of UVMs should increase with an increase in v. However, from a certain value of v, the attack becomes beneficial and m^* becomes positive, which initially causes a drop of k^*. Then k^* increases to compensate the growth of v. m^* also increases with v because the attacker has greater chances to access the entire data, which makes the increase of m^* reasonable. m^* quickly reaches the maximum allowed value of 100. When $n = 30$ (Fig. 10), the attacker either abstains from attack ($m^* = 0$) or attacks with the maximum possible number of AVMs ($m^* = 100$). By increasing the number of UVMs the user tries to deter the attacker from the attack. When both m^* and k^* reach their maximum values, the expected losses become linear functions of v because the rest of parameters in (6) become constant.

6. Conclusion and future directions

Extensive uses of the virtualization technology, particularly, VMs in the cloud system have brought some unique security concerns for the cloud users. This work has focused on one of such concerns, co-resident attacks, where user's sensitive information in one VM can be accessed through side channels by another co-resident VM of malicious attackers. The existing solutions are mostly dedicated to handling side channels or VMs allocations, which often involve modifications of the current cloud system architecture or need actions from the cloud providers. In this work we address the co-residence attacks from a different perspective of user's requests. Specifically, based on the suggested probabilistic

co-residence coverage model and the data partition technique, the optimal number of UVMs requests (one UVM per data block) from a user is determined. Three cases are considered: a fixed number of AVMs, an uncertain number of AVMs, and strategic attackers who choose the number of AVMs maximizing their benefit. As demonstrated through examples the proposed scheme can effectively mitigate effects of the co-resident attacks through minimizing user's losses or costs.

In the future we will consider situations where the success of an attacker's access to user's data in each server with co-resident VMs is independent from such success in other servers using different data protection mechanisms.

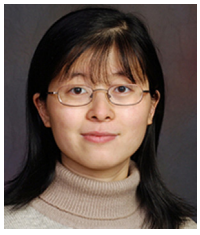
Acknowledgments

This work was supported in part by the Natural Science Foundation of China under Grant 61170042, by the Fundamental Research Funds for the Central Universities under Grant ZYGX2011Z001 and by the Innovational Team Project of Sichuan Province (No. 2015TD0002).

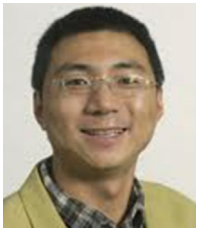
References

[1] A. Beloglazov, J. Abawajy, R. Buyya, Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing, Future Gener. Comput. Syst. 28 (5) (2012) 755–768.
[2] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, M. Zaharia, A view of cloud computing, Commun. ACM 53 (4) (2009) 50–58.
[3] F. Palmieri, U.o Fiore, S. Ricciardi, A. Castiglione, GRASP-based resource re-optimization for effective big data access in federated clouds, Future Gener. Comput. Syst. 54 (2016) 168–179.
[4] R. Buyya, C.S. Yeo, S. Venugopal, J. Broberg, I. Brandic, Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility, Future Gener. Comput. Syst. 25 (6) (2009) 599–616.
[5] T. Ristenpart, E. Tromer, H. Shacham, S. Savage, Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds, in: Proceedings of the 16th ACM conference on Computer and communications security, 2009, pp. 199–212.
[6] M. Godfrey, M. Zulkernine, Preventing cache-based side-channel attacks in a cloud environment, IEEE Trans. Cloud Comput. 2 (4) (2014) 395–408.
[7] H. Hlavacs, T. Treutner, J.-P. Gelas, L. Lefevre, A.-C. Orgerie, Energy consumption side-channel attack at virtual machines in a cloud, in: Proc. of 9th IEEE Int. Conf. Dependable, Auto. Secure Comput. (DASC), Dec. 2011, pp. 605–612.
[8] Y. Han, Defending against co-resident attacks in cloud computing (Ph.D. thesis), University of Melbourne, 2015.
[9] G. Nalinipriya, P.J. Varalakshmi, K.G. Maheswari, R. Anita, An extensive survey on co-resident attack in dynamic cloud computing environment, Int. J. Appl. Eng. Res. 11 (5) (2016) 3019–3023.
[10] M.M. Alani, Securing the cloud: Threats, attacks and mitigation techniques, J. Adv. Comput. Sci. Technol. 3 (2) (2014) 202–213.
[11] J. Shi, X. Song, H. Chen, B. Zang, Limiting cache-based side-channel in multi-tenant cloud using dynamic page coloring, in: Proc. of IEEE/IFIP 41st International Conference on Dependable Systems and Networks Workshops (DSN-W), July 2011.
[12] J. Wu, L. Ding, Y. Lin, N. Min-Allah, Y. Wang, XenPump: A new method to mitigate timing channel in cloud computing, in: Proc. of IEEE 5th International Conference on Cloud Computing (CLOUD), 2012.
[13] T. Kim, M. Peinado, G. Mainar-Ruiz, STEALTHMEM: System level protection against cache-based side channel attacks in the cloud, in: Proc. of 21st USENIX Secur. Symp., 2012.
[14] Y. Zhang, M.K. Reiter, Düppel: Retrofitting commodity operating systems to mitigate cache side channels in the cloud, in: Proc. of ACM SIGSAC Conf. Comput. Commun. Secur. (CCS), 2013, pp. 827–838.
[15] V. Varadarajan, T. Ristenpart, M. Swift, Scheduler-based defenses against cross-VM side-channels, in: Proc. of 23rd USENIX Secur. Symp., 2014, pp. 687–702.
[16] S. Yu, X. Gui, J. Lin, An approach with two-stage mode to detect cache-based side channel attacks, in: Proc. of Int. Conf. Inf. Netw. (ICOIN), 2013, pp. 186–191.
[17] S. Sundareswaran, A. Squcciarini, Detecting malicious co-resident virtual machines indulging in load-based attacks, in: S. Qing, J. Zhou, D. Liu (Eds.), Information and Communications Security, in: (Lecture Notes in Computer Science), : Springer-Verlag, New York, NY, USA, 2013, pp. 113–124.
[18] F. Liu, Q. Ge, Y. Yarom, F. Mckeen, C. Rozas, G. Heiser, R.B. Lee, CATalyst: Defeating last-level cache side channel attacks in cloud computing, in: Proc. of IEEE International Symposium on High Performance Computer Architecture (HPCA), March 2016.
[19] Y. Zhang, A. Juels, A. Oprea, M.K. Reiter, HomeAlone: Co-residency detection in the cloud via side-channel analysis, in: Proceedings of IEEE Symposium on Security and Privacy, IEEE Computer Society, Washington, DC, USA, 2011, pp. 313–328.
[20] A. Bates, B. Mood, J. Pletcher, H. Pruse, M. Valafar, K. Butler, On detecting co-resident cloud instances using network flow watermarking techniques, Int. J. Inf. Secur. 13 (2) (2014) 171–189.
[21] A. Bates, B. Mood, J. Pletcher, H. Pruse, M. Valafar, K. Butler, Detecting co-residency with active traffic analysis techniques, in: Proceedings of ACM Workshop on Cloud computing security workshop, New York, NY, USA, 2012, pp. 1–12.
[22] Z. Xu, H. Wang, Z. Wu, A Measurement Study on Co-residence Threat inside the Cloud, in: Proceedings of the 24th USENIX Security Symposium, Washington, D.C. August, 2015, pp. 12–14.
[23] V. Varadarajan, Y. Zhang, T. Ristenpart, M. Swift, A placement vulnerability study in multi-tenant public clouds, in: Proceedings of the 24th USENIX Conference on Security Symposium, USENIX Association Berkeley, CA, USA, 2015, pp. 913–928.
[24] Y. Han, J. Chan, T. Alpcan, C. Leckie, Virtual machine allocation policies against Co-resident attacks in cloud computing, in: Proc. of IEEE International Conference on Communications (ICC 2014), 2014, pp. 786–792.
[25] Y. Han, J. Chan, T. Alpcan, C. Leckie, Using virtual machine allocation policies to defend against co-resident attacks in cloud computing, IEEE Trans. Dependable Secure Comput. (2015) in press, http://dx.doi.org/10.1109/TDSC.2015.2429132.
[26] Y. Azar, S. Kamara, I. Menache, M. Raykova, B. Shepard, Colocation-resistant clouds, in: Proc. of the 6th ACM Workshop Cloud Comput. Secur., 2014, pp. 9–20.
[27] Y. Han, J. Chan, T. Alpcan, C. Leckie, A game theoretical approach to defend against co-resident attacks in cloud computing: Preventing Co-residence using semi-supervised learning, IEEE Trans. Inf. Forensics Secur. 11 (3) (2015) 556–570.
[28] Y. Han, T. Alpcan, J. Chan, C. Leckie, Security Games for Virtual Machine Allocation in Cloud Computing, in: Proceeding of 4th International Conference on Decision and Game Theory for Security, Vol. 8252, Springer-Verlag, New York, 2013, pp. 99–118.
[29] H.S. Bedi, S. Shiva, Securing cloud infrastructure against co-resident DoS attacks using game theoretic defense mechanisms, in: Proc. of the International Conference on Advances in Computing, Communications and Informatics, ACM, New York, NY, USA, 2012, pp. 463–469.
[30] G. Levitin, K. Hausken, H. Taboada, D.W. Coit, Data survivability vs. security in information systems, Reliab. Eng. Syst. Saf. 100 (2012) 19–27.
[31] A. Amin Soofi, M. Irfan Khan, Fazal-e-Amin, A review on data security in cloud computing, Int. J. Comput. Appl. (0975–8887) 94 (5) (2014).
[32] V. Ganapathy, D. Thomas, T. Feder, H. Garcia-Molina, R. Motwani, Distributing data for secure database services, Trans. Data Priv. 5 (1) (2012) 253–272.
[33] Y. Li, R. Peng, Service reliability modeling of distributed computing systems with virus epidemics, Appl. Math. Model. 39 (18) (2015) 5681–5692.
[34] L. Zheng, S. Chong, A.C. Myers, S. Zdancewic, Using replication and partitioning to build secure distributed systems, in: Proceedings of the 2003 IEEE Symposium on Security and Privacy, 2003, pp. 236–250.
[35] S. AhirraoEmail, R. Ingle, Scalable transactions in cloud data stores, J. Cloud Comput. Adv. Syst. Appl. 4 (2015) 21.
[36] H. Hazila, C. Suriayati, Efficient and secured data partitioning in the multi cloud environment, J. Inf. Assur. Secur. 10 (5) (2015) 200–208.
[37] L. Yang, J. Cao, Y. Yuan, T. Li, A. Han, A. Chan, A framework for partitioning and execution of data stream applications in mobile cloud computing, Newsletter ACM SIGMETRICS Perform. Eval. Rev. 40 (4) (2013) 23–32. ACM New York, NY, USA, March.
[38] Y. Shinde, A. Vishwa, Privacy preserving using data partitioning technique for secure cloud storage, Int. J. Comput. Appl. (0975–8887) 116 (16) (2015).
[39] R. Leistikow, D. Tavangarian, Secure picture data partitioning for cloud computing services, in: Proc. of 27th International Conference on Advanced Information Networking and Applications Workshops, 2013, pp. 668–671.
[40] V.M. Kurandale, P. Dhumal, A. Gobare, K. Gaikwad, Data partitioning technique to improve cloud data storage security, Int. J. Emerg. Technol. Innovative Res. 2 (4) (2015) 917–921.
[41] S. Tirodkar, Y. Baldawala, S. Ulane, A. Jori, Improved 3-dimensional security in cloud computing, Int. J. Comput. Trends Technol. (IJCTT) 9 (5) (2014).
[42] A. Kanade, R. Mule, M. Shuaib, N. Nagvekar, Improving cloud security using data partitioning and encryption technique, Int. J. Eng. Res. Gen. Sci. 3 (1) (2015).
[43] M. Shaikh, A. Achary, S. Menon, N. Konar, Improving cloud data storage using data partitioning and data recovery using seed block algorithm, Int. J. Latest Technol. Eng., Manag. Appl. Sci. 4 (1) (2015).
[44] A. Nadaph, V. Maral, Cloud computing - partitioning algorithm and load balancing algorithm, Int. J. Comput. Sci., Eng. Inf. Technol. 4 (5) (2014).
[45] C. Selvakumar, Improving cloud data storage security using data partitioning technique, in: Proc. of IEEE 3rd International Advance Computing Conference (IACC), 2013, pp. 7–11.
[46] R. Peng, Q. Zhai, G. Levitin, Defending a single object against an attacker trying to detect a subset of false targets, Reliab. Eng. Syst. Saf. 149 (2016) 137–147.

Gregory Levitin is presently a distinguished visiting professor at University of Electronic Science and Technology of China and a senior expert at the Reliability Department of the Israel Electric Corporation. His current interests are in operations research and artificial intelligence applications in reliability, defense and power systems. In this field Prof. Levitin has published more than 250 papers and four books. He is senior member of IEEE and chair of the ESRA Technical Committee on System Reliability. He is member of editorial boards of Reliability Engineering & System Safety, Journal of Risk and Reliability, and Reliability and Quality Performance, served as associate editor of IEEE Transactions on Reliability.



Liudong Xing received the B.E. degree in computer science from Zhengzhou University, China in 1996, and the M.S. and Ph.D. degrees in electrical engineering from the University of Virginia in 2000 and 2002, respectively. She is currently a Professor with the Department of Electrical and Computer Engineering, University of Massachusetts (UMass) Dartmouth, USA. She is also an Adjunct Professor with the Collaborative Autonomic Computing Laboratory, University of Electronics Science and Technology of China. She is an Associate Editor for International Journal of Systems Science and International Journal of Systems Science: Operations & Logistics. She is also an Assistant Editor-in-Chief for International Journal of Performability Engineering. Dr. Xing is the recipient of the 2010 Scholar of the Year Award, and 2011 Outstanding Women Award of UMass Dartmouth and the IEEE Region 1 Technological Innovation (Academic) Award in 2007. She is also the co-recipient of the Best Paper Award at the IEEE International Conference on Networking, Architecture, and Storage in 2009. Her research focuses on reliability modeling and analysis of complex systems and networks.



Yuanshun Dai received the B.S. degree from Tsinghua University, Beijing, China, in 2000, and the Ph.D. degree from the National University of Singapore, Singapore, in 2003. He is Associate Dean of School of Computer Science and Engineering, University of Electronic Science and Technology of China. He is also Chaired Professor and the Director of the Collaborative Autonomic Computing (CAC) Laboratory. He serves as Chairman of the Professor Committee in the School since 2012; and as the Associate Director at the Youth Committee of the "National 1000er Plan" in China. He has published more than 100 papers and 5 books, where there are 50 papers indexed by SCI including 25 IEEE/ACM Transactions papers. His current research interests include Cloud Computing and Big Data, Reliability and Security, Modeling and Optimization. Dr. Dai has served as a Guest Editor of the IEEE Transactions on Reliability. He is also on the editorial boards of several journals.