Design Issues of the Side-Channel Attacks Protecting Scheme in Cloud Computing Environment

Shin-Jer Yang, Chia-Chi Yen

Dept. of Computer Science and Information Management, Soochow University, Taipei, Taiwan
sjyang@csim.scu.edu.tw, monkey15679@gmail.com

Abstract

The computing resources can be utilized and shared with other VMs on the same physical machine, thus there exists information security in cloud computing. Cloud services such as IaaS, PaaS and SaaS can employ the multi-tenancy control to accomplish the applications independence and data isolation for different tenants. The SCA attacker can break into the shared computing resources and steal stored data of other users on the physical machine, which results in data leakage and theft. Therefore, we examine and fix the security issues of current CP-SCA to propose new CRDPS scheme for enhancing defense capability of SCA.

The CRDPS can monitor the ICMP and TCP SYN packets to determine whether the sender is a SCA attacker. Then, we perform some simulations using UNB CIC Dataset to analyze and compare the CRDPS and CP-SCA schemes in terms of four KPIs. Finally, the simulation results indicate that the CRDPS has a better detection rate, higher accuracy ratio, and system throughput than the CP-SCA about 8.51%, 41.36%, and 251 packets respectively, but there is a 4.28% overhead in average processing time. Consequently, the proposed CRDPS can accurately identify the attackers to harden the security and enhance the total quality in cloud services, especially in SaaS.

Keywords: Cloud computing, Side-Channel Attacks, ICMP, TCP SYN, Co-Residency Detection

1 Introduction

Since the cloud computing technology is the virtual machines (VMs) that to be formed through virtualization based on the sharing of computer resources [1], the information security concerns come up. Cloud computing services can be divided into three levels: IaaS (Infrastructure as a Service), PaaS (Platform as a Service), and SaaS (Software as a Service). Cloud computing services can be used by multi-tenancy control, for different tenant's application environment isolation such like application context independence and data isolation. Hence, the multi-tenancy control is to ensure that the different tenants' applications will not interfere with each other, while the confidentiality of information is also strong enough. An attacker can steal the computing resources on the shared physical machine through the Side-Channel Attack (SCA), who may even steal the data which other users have saved or retrieved on the physical machine. SCA obtains the targeted private information through shared hardware resource analysis on the attack target, while Co-Residency Detection (CRD) should be performed before the SCA is launched. The attacker will verify whether the attack target is on the same physical machine, and the SCA can be launched only after the confirmation. The three steps of information security are Prevention, Detection, and Action. Since that the prevention is better than detection or action, the system can also support the access control lists of user applications by utilizing the RB-MTAC to prevent and avoid the occurrence of SCA, when the CRD behavior is detected. The current CP-SCA (Conventional Protection for Side-Channel Attack) emphasizes making the CRD analysis prior to the Side-Channel Attack, to monitor the ICMP packets required for the CRD for traffic statistics, analyze the transmission of its packets, and identify suspicious attackers and victims in the cloud environment [2]. However, in addition to using ICMP packets for confirming whether co-utilized physical machines are used, the CRD can also carry out the confirmation through TCP SYN packets. Therefore, this paper proposes the CRDPS (Co-Residency Detection with Packet Sniffer) scheme to improve the SCA issues of current CP-SCA for monitoring the ICMP and TCP SYN packets into consideration to upgrade the defense capability.

With the vigorous development and popularization of cloud computing technology, the problems concerning information security in its applications have gradually become a subject that deserves in-depth investigations. In particular, cloud computing is mainly to share computing resources, or even storage spaces through virtualization. If the attackers have attacked through SCA, the private messages that are stored in the cloud could be stolen or misappropriated. The

*Corresponding Author: Shin-Jer Yang; E-mail: sjyang@csim.scu.edu.tw
DOI: 10.3966/160792642020052103009

primary requirement for SCA is that the cloud environment of the attacker and what of the victim to be attacked are both on the same physical machine. To make sure both are on the same physical machine, the attacker will first perform CRD to detect whether the attack target is on the same physical machine, and further attacks will be made after the confirmation. If the attacker can be detected in the process of CRD, the SCA can be prevented from occurring accordingly. Therefore, the proposed CRDPS of this paper are to continue the existing researches on SCA's prevention schemes, and consider further on the possibility of monitoring through TCP SYN packets. Hence, the main purposes of this paper are as follows:

- In the cloud computing environment, side-channel attack is a newer security threat. This paper continues using the CP-SCA, but it needs to fix its drawback that it merely detects the ICMP by designing a new CRDPS approach.
- To achieve more accurate judgment results through the detections of ICMP packets and the monitoring on the TCP SYN packets.
- The occurrences of side channel attack can be alleviated through the proposed CRDPS scheme under any cloud services.
- According to the UNB CIC Dataset, we can set up a simulation system to generate packets to emulate the packets produced by CRDs for SCA attacker on the cloud virtual machines.
- We can perform some simulations using emulated packets to be produced by UNB CIC Dataset to analyze and compare the CRDPS and CP-SCA methods in terms of four key performance indicators (KPIs).
- In summary, the proposed CRDPS can provide safer and more secure quality of service than CP-SCA under the cloud environment.

The remainder of this paper is organized as follows. In Section 1, this paper introduces the research backgrounds and purposes. In Section 2, we survey and describe the reviews of related works and current security issues under multi-tenant architecture. Also, we illustrate identity management and access control issues of Side-Channel Attacks in multi-tenancy control. Section 3 examines operations of the CRDPS scheme and designs its algorithm. Section 4 comprises the set of simulation experiments for analyzing the results in terms of four KPIs. Section 5 comprises the set of simulation experiments and analyzes the final simulation results. Then, we draw some conclusions and indicate the future research directions in Section 6.

2 Related Works

2.1 Cloud Computing

There is an essential technology in Cloud computing, that is hardware virtualization. The physical machine can be virtualized to form some virtual machines via the virtualization. In other words, there are many virtual machines residing on a physical machine, and therefore the virtual machines will share the computing resource of the physical machine, such as CPU, memory, and disk space. In the National Institute of Standards and Technology (NIST), "cloud computing" is a model that is convenient and able to tune and access to a wide range of shared computing resources such as servers, storage, network, applications, services, etc. according to user on-demand [3]. Also, it can simplify the management of the work and the interaction with the cloud service providers to support three kinds of services rapidly and elastically.

2.2 Multi-tenancy Control

Cloud computing services can be divided into three levels: IaaS (Infrastructure as a Service), PaaS (Platform as a Service) and SaaS (Software as a Service). Then, IaaS providers have recently been building marketplaces of "cloud apps," which are VMs pre-installed with a variety of software stacks. This allows clients to deploy services such as network middle-boxes on their work VMs without the cloud provider [4]. The security requirement is more relevant in PaaS, The platform must ensure that no malicious or faulty code from any tenant can interfere with the normal execution of other users' code or with the platform itself [5]. Also, SaaS adoption presents security risks. Moving a company's sensitive data into the cloud providers, which expand and complicate the risk in organizational operations [6].

The cloud services can support for multiple users under multi-tenant architecture, hence the multitenancy control can make various users have designated roles, and different roles have respective functions and permissions in cloud services. In previous research, the RB-MTAC is mainly to integrate the identity management and role-based access control for different tenant applications under the multi-tenant cloud environment [7]. That is to ensure data isolation and make different tenant's applications not interfere with each other, while the confidentiality of information is also strong enough.

2.3 Side-Channel Attacks

A side channel attack is a non-intrusive attack. It does not go through violent crack (password crack) or utilize the vulnerabilities of algorithms. Instead, it is undertaken by the way of observing the hardware resources operation, analyzing, and then proceeding with the side channel attacks [8].

Side channel attacks occur under the situation with a shared CPU, shared cache, shared computing units and other shared hardware resources. For attacks on shared CPU resource, an attacker can find out the scheduling vulnerability of the Amazon EC2, which can exploit

CPU in computing resources originally owned by other users [9]. For cache-sharing attack, an attacker can observe the executions of different instructions or actions to obtain the target caching resources [10-12].

Except above these two side-channel attack modes, an attacker can acquire the related target information by observing the execution time between instructions in a system for the latency time attack [13-14]. There would be a specific behavior before an attack is taking place, and that is the Co-Residency Detection (CRD). In other words, the attacker needs to perform CRD to confirm whether the attack target is on the same physical machine before proceeding with the attack on the side channel. The side channel attacks can proceed only after the confirmation is finished, the SCA process is as shown in Figure 1. Therefore, we can identify the occurrence of side channel attacks through monitoring the CRD behaviors.

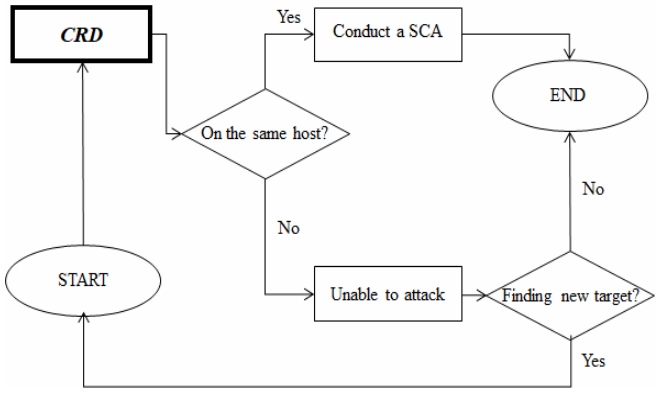

Figure 1. Side-Channel Attack flow

2.4. Co-Residency Detection

The Co-Residency Detection (CRD) is used for confirming whether two virtual machines are on the same physical machine. The CRD can be performed via sending ICMP and TCP SYN packets for Traceroute, and then the IP address of the Dom0 virtual machine can be obtained. If the two Dom0 IP addresses are the same, and then it indicates that the two virtual machines share the same physical machine. Then, we can monitor ICMP and TCP SYN packets to identify the occurrence of CRD behaviors and can tell whether a SCA would be happen after the statistical analysis. In summary, the Traceroute functions of ICMP and TCP SYN control packets are as follows.

2.4.1 ICMP Traceroute

The program sends out a UDP packet with TTL = 1. The first router subtracts the TTL by 1 to get to 0, and then no longer continues to forward this packet. Instead, it returns with an ICMP timeout packet [15]. The retrieving side thus may obtain the address of the first gateway, which the timeout packet has gone through. Then a TTL = 2 packet is sent out again to obtain the 2nd gateway address, thus all the addresses on the gateway may be obtained by serially by increasing the TTL. Not all the gateways return the timeout packets, which is what actually happens, for security reasons. Most firewalls or firewall-activated routers have been configured as not to respond to all kinds of ICMP packets, therefore the Traceroute procedure does not necessarily get all the router gateway addresses along the way.

2.4.2 TCP Traceroute

The traditional Traceroute uses ICMP packets to get the destination path. Nowadays firewalls have been commonly used in networks that many packets sent by Traceroute are filtered out so that they cannot track all the way to the destination. However, in many cases these firewalls allow certain TCP port packets to enter the back-end hosts. By sending TCP SYN packets instead of ICMP packets, the TCP SYN Traceroute is able to pass the filtering criteria of most of the firewalls [16].

Cloud computing technology forms multiple virtual machines through virtualization based on shared computing resources. Most cloud services are set up on Multi-Tenant Architecture (MTA), which allows modeling and exploiting efficiently huge amounts of computing resources. Hence, computing resources to be used may share the same physical machine with other virtual machines, which can be supervised by multi-tenancy control to conduct the application independence as well as the isolation of data among different tenants. However, the data or application on the computing resources can break into another virtual machine by SCA. Before performing SCA, determine whether the target of an attack is on the same physical machine first, therefore, the detection of CRD is required to be conducted via Traceroute. After confirming the attack target is on the same physical machine, an SCA attacker will launch the attacks.

3 Operations and Dessign Issues of Crdps

This paper proposes CRDPS and to integrate with the SYN TCP packet detection mechanism based on the current CP-SCA. Through the CRDPS scheme, the ICMP and TCP SYN packet flows can be both identified to determine whether the packet sender is an SCA attacker. It can enhance the defense of side-channel attacks. The CRDPS scheme has the following characteristics.

- To determine whether a virtual machine user is conducting a side-channel attack, and if an attacker is identified, put it into the illegal IP addresses list.
- A proposed defense scheme in side channel attacks can be applicable to different cloud service models.
- The defense capabilities of SCA can be enhanced through comparing ICMP and TCP SYN packets.

3.1 Operation Structure and Description

The attacker can trigger a CRD upon the target to confirm whether it is on the same physical machine, and the SCA can be performed after the confirmation. The CRDPS figures out the illegal IP addresses list through monitoring ICMP and TCP SYN packets. Hence, the operations of the SCA process are as shown in Figure 2.

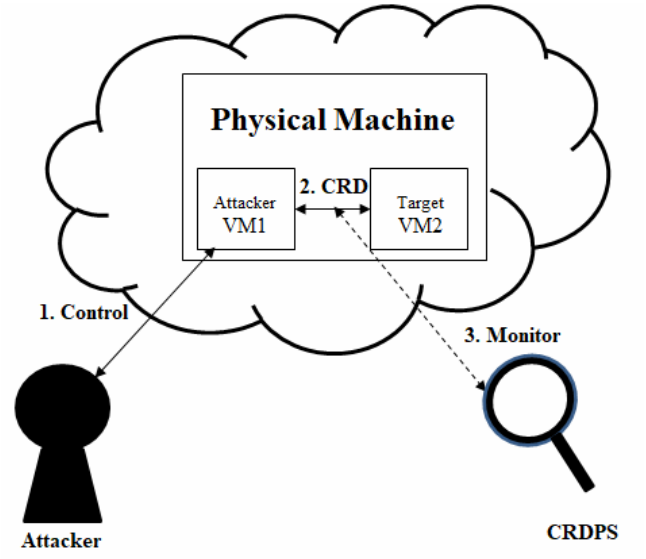

Figure 2. SCA operations structure

This paper calculates the repeated thresholds of IP addresses and finds out the illegal IP address through the CRDPS scheme. The detection process is as shown in Figure 3, and the operational flows are illustrated as following.

(a) The CRD packet log files captured by Wireshark can read into the CRDPS program.

(b) The ICMP and TCP SYN packets can be filtered from the various types of packets through the CRDPS, and then be stored into the data tables of the CRDPS.

(c) To count the number of source IP Addresses to be filtered from the ICMP and TCP SYN packets.

(d) Calculating the threshold of repetition of the source IP addresses.

(e) If the repetition of source IP Address is greater than the threshold, put it into the list of illegal IP Addresses. And if the repetition of source IP Address is not greater than the threshold, put it into the list of legal IP addresses.

(f) To identify the list of legal or illegal IP Addresses through the CRDPS.

(g) To determine whether to continue retrieving any packets or not.

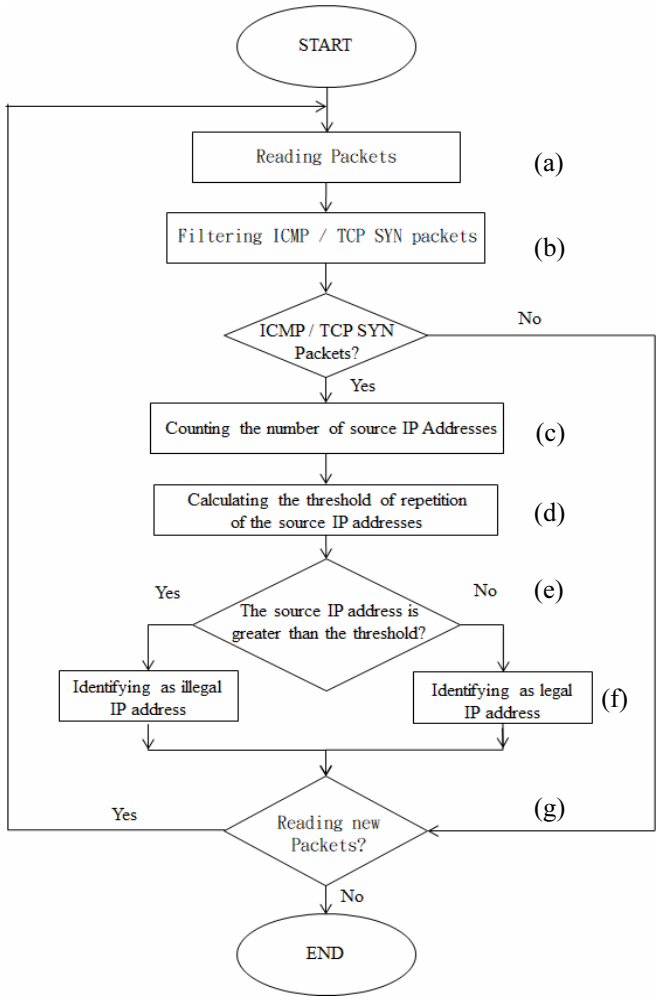

Figure 3. Detection Activities in CRDPS

3.2 Algorithm Design

Based on operations of Figure 3, the algorithm of CDRPS can be designed as follow:

Algorithm CRDPS()

Input:
String[] CRDPS;
String[] Suspicious;
int[] IPSourceCount;
int Threshold_Val**;**
bool ifPacketsInput = true;
bool ifAttacker
Output:
To complete CRDPS Method for resolving Cloud SCA issues
Method:

```
1.  BEGIN {
2.    While(ifPacketsInput) {
3.      Load_CRDP();                    //(a)
4.        Packets_Filter();             //(b)
5.      IP_Source_Count();              //(c)
6.      Threshold_Cal();                //(d)
7.      ifAttacker = Check_CRDPS();     //(e)
8.      if(ifAttacker == true){
```

```
9.          List_ Illegal _IP();                //(f)
10.     }else{
11.       List_Legal_IP();                      //(f)
12.     }
13.     ifPacketsInput = Input_Check();         //(g)
14.   }
15. } END
16. Procedure Load_CRDPS() {
17.   String crdps = getCRDPS();
18.   CRDPS = crdps.Split(‘ ’);
19. } END Load_CRDPS
20. Procedure Packets_Filter() {
21.   Suspicious = ifICMP_TCPSYN();
22. } END Packets_Filter
23. Procedure IP_Source_Count() {
24.   IPSourceCount++;
25. } END IP_Source_Count
26. Procedure Threshold_Cal() {
27.   Threshold_Val = IPSourceCount.getAverage();
28. } END Threshold_Cal
29. Procedure Check_CRDPS() {
30.    If (IPSourceCount > Threshold_Val){
31.      return true;
32.      }else{
33.      return false;
34.    }
35. } END Check_CRDPS
36. Procedure List_Illegal () {
37.   addUserToBlacklisting();
38. } END List_ Illegal
39. Procedure List_Legal () {
40.   addUserToWhitelisting();
41. } END List_Legal
42. Procedure Input_Check() {
43.   if(ifAnyInput()){
44.   return true;
45.   }else{
46.    return false;
47. }
48. } END Input_Check
END CRDPS.
```

4 Simulations Environments and KPIs

In this paper, we conducted simulation experiments to compare and analyze between CRDPS and CP-SCA methods in terms of four key performance indicators (KPIs) such as detection rate, accuracy ratio, system throughput, and average processing time.

4.1 Experimental Environment Configuration

Initially, we set up an experimental environment to simulate the behaviors of CRD and to generate emulated packets. Our simulation operation architecture is based on above Figure 2, that is the attacker’s virtual machine (VM1) and the target’s virtual machine (VM2) are resided on the same physical machine. Then, the legal or illegal IP address can be identified by the CRDPS scheme as shown in Figure 4. First, the CRD behaviors are simulated through the built-in DOS commands of Windows to create attacking packets based on the operations of the UNB CIC Dataset [17]. These packets of CRD behaviors can be fetched by Wireshark. We first calculate the average number of packets produced by one virtual machine and then simulate the packets generated by multiple virtual machines. Next, the ICMP and TCP SYN packets can be filtered by the proposed CRDPS scheme in this paper to make a statistics of the number of repeated IP addresses, and the IP address over the threshold are identified as illegal IP address. The threshold value of this paper is taken as the average of the number of packets from repetitive IP addresses. The legal IP address may be misjudged as illegal IP, if the threshold value is lower, thus resulting in a rising in number of misjudged packets; if the threshold value is higher, as a result, the illegal IP addresses are skipped and the number of missed leaking packets is too high. Therefore, the averaging threshold value is used, but this threshold value can be flexibly tuned according to the actual needs in the cloud environment. Hence, the system specifications for the implementations of simulation environments are listed in Table 1 and Table 2.

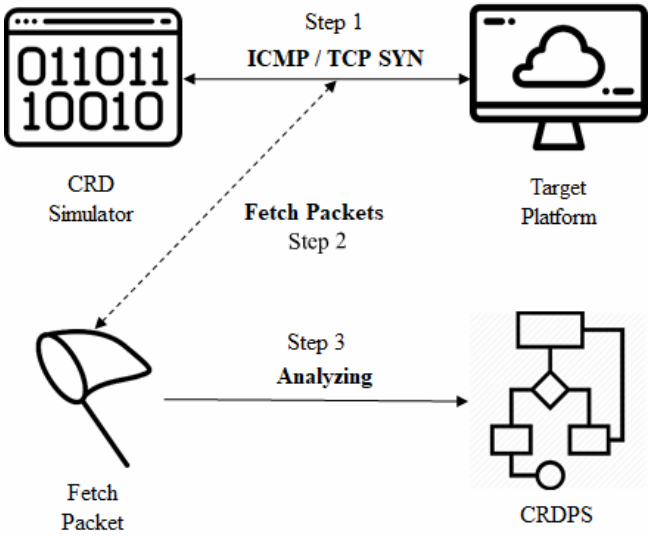

Figure 4. Simulations environment

Table 1. Physical machine specifications

Software / Hardware	Specification
Operation System	Windows 10
CPU	4 Cores
Memory	16GB
Disk	256GB

Table 2. Virtual machines specifications

Software / Hardware	Specification
Operation System	Windows 7
CPU	2 Cores
Memory	2GB
Disk	30GB

4.2 KPIs Definitions and Descriptions

For simulation experiments, there are four KPIs including detection rate, accuracy ratio, system throughput, and average processing time as shown in Table 3. Also, all KPIs as shown in Table 3 are computed as Formulas (1) to (4).

Table 3. The definitions of KPIs' list

Key Performance Indicators (KPIs)	
KPIs	**Purposes**
Detection Rate (**DR**)	Malicious packets launched by attackers can be detected by CRDPS and CP-SCA schemes to reduce the probability of being attacked
Accuracy Ratio (**AR**)	The accuracy ratio is determined through the calculation to differentiate the filtering correctness of CRDPS and CP-SCA schemes.
System Throughput (**ST**)	The number of illegal packets that can actually be handled within a fixed period of time (5 min).
Average Processing Time (**APT**)	The average processing time is to calculate the averaging of *N* operations time of CRDPS and CP-SCA schemes.

$$DR = \frac{\text{Abnormal Packets}}{\text{Total Packets}} \quad \textbf{(1)}$$

$$AR = \frac{\text{Total Illegal IP Address Packets}}{(\text{The Actual Illegal Packets} + \text{The Misjudged Packets})} \quad \textbf{(2)}$$

$$ST = \frac{\text{Total Illegal IP Address Packets}}{\text{Fixed Period of Time (5min)}} \quad \textbf{(3)}$$

$$APT = \frac{\text{Total Processing Time}}{N} \quad \textbf{(4)}$$

5 Simulations Results and Analysis

This paper refers to the UNB CIC Dataset how to generate attacking packets [17]. The operational flow of the UNB CIC Dataset to create attacking packets includes complete network configuration, collecting normal and abnormal packets. First, we built a virtual machine to collect the packets under its normal operation. Next, we performed simulations on the CRD behaviors and collected the ICMP and TCP SYN packets generated by an SCA attacker for CRD. These packets recorded normal situations and the packets were generated during the CRD every five minutes. A total of ten simulations were logged and recorded by the system. Finally, we calculated the normal situations and the average number of packets produced via the CRD every five minutes.

The number of packets generated by the CRD performed by the attacker in the case of building multiple virtual machines is simulated. According to the proposed CRDPS and the CP-SCA methods, the simulation experiments are conducted respectively, although the CRDPS method obtains a 4.28% overhead on average processing time to be compared with the CP-SCA. In addition, the other KPIs are improved significantly. The detection rate, accuracy ratio, system throughput, and average processing time are illustrated as below in detail.

5.1 Detection Rate

The ICMP and TCP SYN packets are filtered from the collected packets to calculate the ratio of the abnormal packets in the collected packets of CRDPS and CP-SCA of this paper, respectively. The simulation results of ten experiments are recorded to indicate that the proposed CRDPS has a higher detection rate than CP-SCA, as shown in Figure 5.

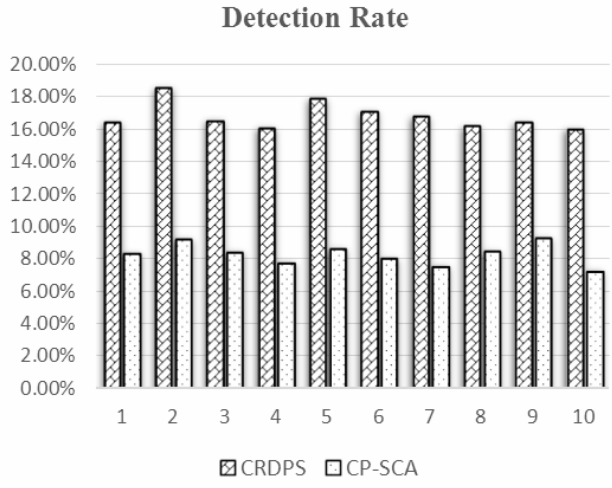

Figure 5. Comparisons in detection rate

5.2 Accuracy Ratio

The ICMP and TCP SYN packets were selected from the collected packets to calculate the ratio of the number of determined illegal IP addresses to account for the total number of IP addresses of attackers between CRDPS and CP-SCA schemes respectively. The simulation results of ten experiments are recorded to present that the proposed CRDPS has a higher accuracy ratio than CP-SCA, as shown in Figure 6.

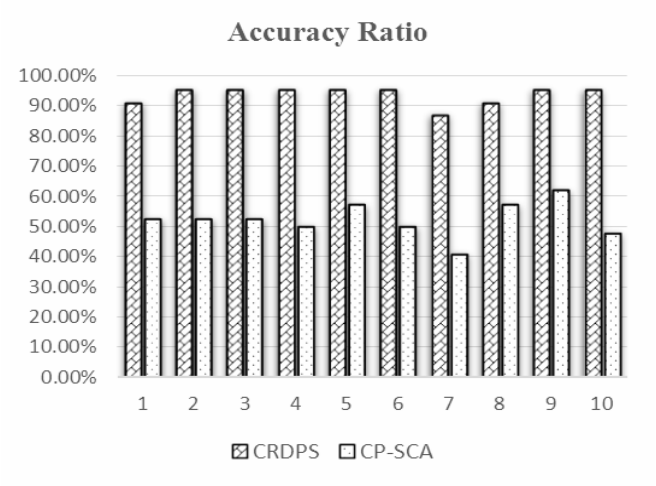

Figure 6. Comparisons in accuracy ratio

5.3 System Throughput

The number of illegal packets that can actually be handled within a fixed period of time, and calculations of the system throughput are made by using CRDPS and CP-SCA schemes, respectively. The simulation results of ten experiments are recorded to demonstrate that the proposed CRDPS has a higher system throughput than CP-SCA, as shown in Figure 7.

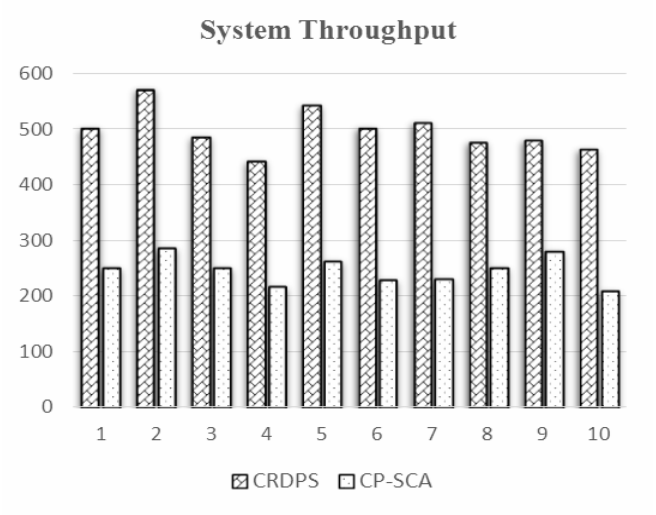

Figure 7. Comparisons in system throughput

5.4 Average Processing Time

The processing time of CRDPS and CP-SCA mentioned in this paper are calculated. The results of ten experiments are recorded to show the average processing time of CRDPS is more consuming 4.28% on average than CP-SCA, as shown in Figure 8.

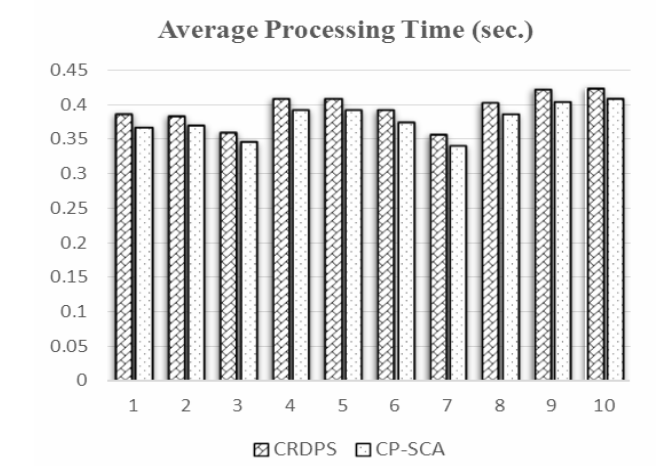

Figure 8. Comparisons in average processing time

This paper summarizes the simulation results in KPI values between CRDPS and CP-SCA schemes to be running in ten experiments as shown in Table 4. In terms of detection rate, the detection rate to be obtained using CRDPS is higher than the CP-SCA, increasing up to 8.51% on average. In terms of accuracy ratio, the accuracy ratio of CRDPS is maintained at 93.55% on average, compared to the CP-SCA, the averaging growth rate is 41.36%. In terms of system throughput, the system throughput with the CRDPS is 497 packets, 251 more packets than the CP-SCA. In terms of average processing time, the proposed CRDPS may have more overhead than CP-SCA about 4.28%. Since the filtering of TCP SYN packets need to be considered, the average processing time may be slightly higher than that of CP-SCA and the usage of computing resources will also be relatively consumed. With regard to reducing average processing time, the computing resources of virtual machines can be improved efficiently and effectively by using Docker technology. The Docker is a new kind of virtualization technology, it can package systems, programs, and execution environments into Containers. Hence, the Docker needs very low system resource requirements and allows multiple systems or applications to be run on the same host simultaneously, thereby increases the utilizations of computing resources.

Table 4. The summarized KPIs values

Experimental results			
KPIs	**CRDPS**	**CP-SCA**	**Difference value**
Detection Rate	16.79%	8.28%	8.51%
Accuracy Ratio	93.55%	52.19%	41.36%
System Throughput	497 Packets	246 Packets	251 Packets
Average Processing Time	0.3944s	0.3782s	-4.28%

6 Conclusion

Cloud computing technology forms multiple virtual machines through virtualization based on shared computing resources. The computing resources to be used may share the same physical machine with other virtual machines, which can be supervised by multi-tenancy control to conduct the application independence as well as data isolation among different tenants. However, the data or application on the computing resources can be broken into another virtual machine by SCA. Hence, this paper proposes a new approach of side-channel defense of CRDPS to fix the CP-SCA drawbacks. The CP-SCA only considers attackers that can conduct CRD via ICMP packets, but it does not consider the TCP SYN packets. In addition to monitoring ICMP packets and incorporating TCP SYN packets into CRDPS scheme, the packets generated by these two CRD behaviors are considered to improve the detection rate, accuracy ratio and system throughput for the side-channel defense mechanism.

According to experimental results, the proposed CRDPS scheme has an additional 4.28% of average processing time than CP-SCA, but a better detection rate and higher accuracy ratio with 16.79%, 8.28%, and 93.55%, 52.19%, respectively. Also, the system throughput is 497 Packets and 246 Packets respectively. Consequently, the final simulation results indicate that the proposed CRDPS is better than CP-SCA in terms of detection rate, accuracy ratio, and system throughput. Thus, the CRDPS can enhance the security and improve the total quality of cloud service models, especially in SaaS. Because of more complexity in CRDPS scheme, so it will consume more computing resources to process the CRDPS. Although we can't tune that the processing time of CRDPS is better than the CP-SCA, we can still shorten the total processing time using the Docker. The future research direction will further enhance the utilizations of computing resources using any virtualization technologies.

Acknowledgements

The partial work of this paper is funded and supervised by the Ministry of Science and Technology in Taiwan under Grant MOST 107-2410-H-031-040-.

References

[1] S. J. Yang, http://www.uis.com.tw/edm/uisnews/uisnews/042/learning.aspx.

[2] Y. H. Wang, *A Mechanism to Prevent Side Channel Attacks in Cloud Computing Environments*, Master's Thesis, Feng Chia University, Taichung, Taiwan, 2014.

[3] P. Mell, T. Grance, *The NIST Definition of Cloud Computing*, Special Publication 800-145, September 2011.

[4] H. Nguyen, V. Ganapathy, A. Srivastava, S. Vaidyanathan, Exploring Infrastructure Support for App-based Services on Cloud Platforms, *Computers & Security*, Vol. 62, pp. 177-192, September, 2016.

[5] L. Rodero-Merino, L. M. Vaquero, E. Caron, A. Muresan, F. Desprez, Building Safe PaaS Clouds: A Survey on Security in Multitenant Software Platforms, *Computers & Security*, Vol. 31, No. 1, pp. 96-108, February, 2012.

[6] C. Tang, J. Liu, Selecting a Trusted Cloud Service Provider for Your SaaS Program, *Computers & Security*, Vol. 50, pp. 60-73, May, 2015.

[7] S. J. Yang, P. C. Lai, J. Lin, Design Issues of Role-Based Multi-Tenancy Access Control in Cloud Computing Services, *Journal of Internet Technology*, Vol. 18, No. 6, pp. 1407-1417, November, 2017.

[8] F. Zhou, M. Goel, P. Desnoyers, R. Sundaram, Scheduler Vulnerabilities and Coordinated Attacks in Cloud Computing, *Computer Security*, Vol. 21, No. 4, pp. 533-559, July, 2013.

[9] D. Page, Defending against Cache-based Side-channel Attacks, *Information Security Technical Report*, Vol. 8, No. 1, pp. 30-44, March, 2003.

[10] D. Page, *Theoretical Use of Cache Memory as a Cryptanalytic Side-channel*, IACR Cryptology e-Print Archive, pp. 169, November, 2002.

[11] E. Tromer, D. A. Osvik, A. Shamir, Efficient Cache Attacks on AES, and Countermeasures, *Cryptology*, Vol. 23, No. 1, pp. 37-71, January, 2010.

[12] J. F. Dhem, F. Koeune, P. A. Leroux, P. Mestré, J. J. Quisquater, J. L. Willems, A Practical Implementation of the Timing Attack, *International Conference on Smart Card Research and Advanced Applications*, Louvain-la-Neuve, Belgium, 1998, pp. 167-182.

[13] R. Hund, C. Willems, T. Holz, Practical Timing Side Channel Attacks against Kernel Space ASLR, *IEEE Symposium on Security and Privacy*, Berkeley, CA, USA, 2013, pp. 191-205.

[14] T. Ristenpart, E. Tromer, H. Shacham, S. Savage, Hey, You, Get off of My Cloud: Exploring Information Leakage in Third-party Compute Clouds, *Proceedings of 2009 16th ACM Conference on Computer and Communications Security*, Chicago, USA, 2009, pp. 199-212.

[15] G. S. Malkin, *Traceroute Using an IP Option*, RFC1393, January, 1993.

[16] L. Witek, TCP Traceroute, http://simulatedsimian.github.io/tracetcp.html.

[17] UNB, CIC IDS 2017 dataset, http://www.unb.ca/cic/datasets/ids-2017.html.

Biographies

Shin-Jer Yang is currently a full Professor in the Department of Computer Science and Information Management, Soochow University, Taipei, Taiwan. Professor Yang is the author/coauthor of more than 120 refereed technical papers (Journals and Conferences)

on Wired/Wireless Networking and Applications, Cloud Computing Applications and Services, Internet Technologies and Applications, and Network Management and Security. Also, he takes in charge of more than 30 research projects. His research interests include Wired/Wireless Networking Technologies and Applications, Cloud Computing and Applications, IoT Applications, Network Management and Security, and Information Management.

Chia-Chi Yen. Currently, He is a Research Assistant in the Department of Computer Science and Information Management, Soochow University, Taipei, Taiwan. His research interests include Cloud Computing and Service, Information Security, and Web Applications Design.