International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:9, No:7, 2015

# Design and Implementation of a Memory Safety Isolation Method Based on the Xen Cloud Environment

Dengpan Wu, Dan Liu

***Abstract*—**In view of the present cloud security problem has increasingly become one of the major obstacles hindering the development of the cloud computing, put forward a kind of memory based on Xen cloud environment security isolation technology implementation. And based on Xen virtual machine monitor system, analysis of the model of memory virtualization is implemented, using Xen memory virtualization system mechanism of super calls and grant table, based on the virtual machine manager internal implementation of access control module (ACM) to design the security isolation system memory. Experiments show that, the system can effectively isolate different customer domain OS between illegal access to memory data.

***Keywords*—**Cloud security, memory isolation, Xen, virtual machine.

## I. Introduction

WITH the development of technology, in recent years there has been a large number of virtualization technology with excellent performance. Xenis one of the representative, an open source virtual machine monitor (VMM or Hypervisor) developed by University of Cambridge [1]. Xen run directly on the physical hardware, and provide the virtual environment operating system can run upward, called domain. VMM has the highest privilege level, control running on the domain. Xen has a privileged domain called the Domain0 domain, the other is called DomainU. Domain0 control hardware devices and provide DomainU management interface for the user. Domain0 is the first virtual domain operation after Xen startup, plays an important role in the system, one side it has management control function, providing a user interface for users; on the other hand it is the device driver domain, running all the hardware device driver. The client Xen support semi-virtualization machine and full virtualization machine. In semi-virtualization mode, the client needs to modify the operating system source code, the Hypercall interface provided by Xen to complete the privileged operation; Full virtualization does not require modifying the client code, but the need for hardware virtualization support CPU [2], mainly used for Windows closed source operating system.

Although Xen is open source and efficient characteristics of [3], but there is also Xen [4] security in their own, such as read sensitive information an attacker can occupied from the guest virtual machine's memory, and can be spread to all attack virtual domain virtual machine management domain VM0 and its upper. Isolation is an important means to ensure the safety and reliability of the virtual machine, the existing virtual machine isolation mechanism mainly include: logical isolation mechanism based on access control; let each virtual machine cannot break through the virtual machine manager gives the virtual by hardware resource constraints; memory protection mechanisms provided by the hardware; the protection mechanism of process address space. This paper through the memory management mechanism of Xen virtual machine, this paper puts forward a security memory isolation method, By the method of grant table access to Xen interception, then complete the validation for the authorization by extending the ACM control, when validation through before the operations of release, finally through the security isolation to control authorization table memory between virtual machines.

Dengpan Wu and Dan Liu are with the University of Electronic Science and Technology of China, Cheng Du, Si Chuan, China (e-mail: ppwuwork@gmail.com, liudan@uestc.edu.cn).

## II. The Design Principle and System

### *A. Xen Memory Management Mechanism*

The memory resource of the computer system is one of the very frequent resource access, Xen uses Linux management ideas in memory virtualization mode, physical memory is divided into a piece of memory, then using the similar structure to the page directory, page table will be memory mapped [5] to the virtual machine memory. Virtual machine manager Hypervisor [6] Xen it will not only be able to its memory mapped to Guest OS space to reduce the TLB (fast table) refresh enormous cost brings, but also can achieve zero copy in the DomainU and Domain0 memory data exchange, in order to improve the efficiency of.

For Xen, one of the most important in the process of memory mapping is the authorization process. Xen memory mapping and memory transfer called via an authorization form Grant Table mechanism to implement. Grant Table mechanism is an efficient mechanism of data transmission of Xen virtualization system, through the Grant Table mechanism can transfer data between different Domain domains, Grant Table mechanism to realize transfer operation [7] mapping operation and page of the page. Xen system to support Grant Table Domain for the realization of the shared memory [8]. Each Domain has its own Grant Table, it is a data structure shared with Xen: it allows Domain to tell Xen other Domain can with what permissions to access the shared page. Table Grant in Table with GR (Grant Reference) said. GR is an integer, represented in the Grant Table index, GR recipient ability to put it as the operating memory. This technique allows shared memory between non-privileged level Domain. The use of Grant Table shared memory access mainly through the following Hypercall (super call) command to:

a. Grant foreign access: The assignment of a list item in the Grant Table, and fill in the corresponding access, when users use the list item, Xen will query access.
b. End foreign access: Check whether GR is being used, then remove the page mapping authority. This operation will mapping prevent future occurrence, but not mandatory recycling existing mapping.
c. Grant foreign transfer: The assignment of a list item in the Grant Table, and specify the recipient authority, when the recipient (Grantee) send a page to the grantor (Granter), Xen will find this table.
d. End foreign transfer: Delete permissions in order to prevent future page transfer occurs, if the transfer has been completed, the Grant Table modification does not affect it.

Grant Reference is accomplished by the following Hypercall, which takes 3 parameters: granttableOP (unsigned int cmd, void* uop, unsigned int count), cmd ordered a Grant Table operation, uop is an operation to describe the structure of the pointer, count indicates that Grant Table the number of operations.

*B. Design Principle*

According to the analysis of Xen memory management mechanism, we can see that the Xen in the memory management mechanism of traditional Linux management has been more mature. It can be safe and stable to complete the allocation and management of virtual machine memory. But the Xen memory in the process of mapping Grant Table mechanism for authorization mechanism between different virtual domain does not take into account the safety requests, when used as a privileged domain running Domain0 (Linux) did not carry out a security check to access memory for other virtual machine, the existence of security risks. Therefore, Xen memory safety isolation, the key lies in the Xen authorization Grant Table memory. Throughout the whole process of Xen Grant Table memory mapping table, can be divided into the following steps:

1. Domain A to create the GR (Grant Reference);
2. Sending GR to Hypervisor;
3. Hypervisor mapped memory;
4. Domain B access memory mapping;
5. After the end of the visit, Domain A recovery GR.

The above steps can be obtained as long as the DomainA need to access any of the DomainB memory, can be Hypervisor mapping. Therefore, we need to add access control in the process of authorization table mechanism. In the Domain to send the GR request, Hypervisor will need to initiate the sender of GR access authentication, only the normal mapping operation is allowed, any abnormal GR request will be terminated. Add a memory mapped GR access control access sequence diagram as shown in Fig. 1.

Using the improved GR authorization, any Xen authorization table operations are audited mechanism. With the help of Xen based security framework provided by ACM, and carries on the expansion of its normal operation, declined to untrusted GR operation, can achieve the security isolation of memory resource target.

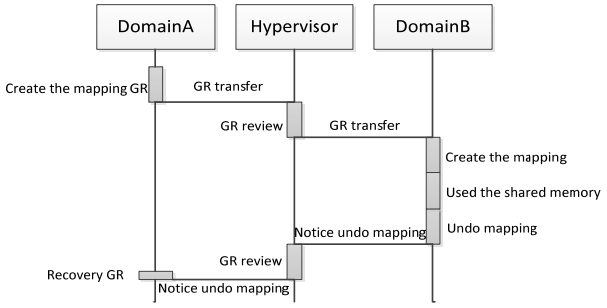


Fig. 1 Memory mapping process improved

*C. The Overall Architecture*

Memory safety isolation method proposed in this paper is implemented in Hypervisor, the extension of the modules in the ACM. Realize the automatic generation of virtual machine memory tracking and ACM rules, security control of Grant Table is realized by HOOKS. The system architecture of the isolation method as shown in Fig. 2:

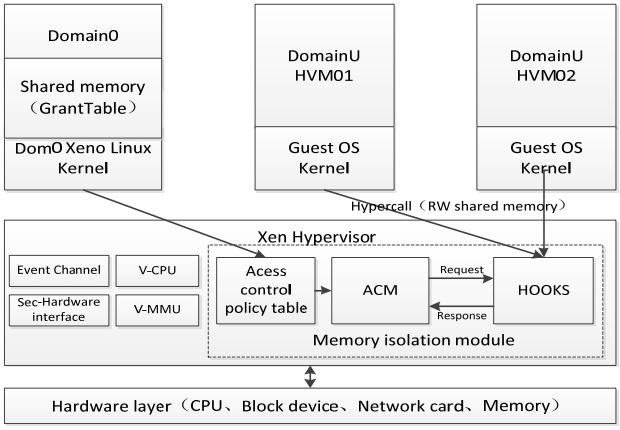


Fig. 2 Memory safety isolation system

When DomainU to access the Domain0 shared memory that created by the Grant Table, the monitored DomainU and Xen was through Hypercall [9] to interaction. By adding HOOKS in the mechanism of Hypercall, can be access behavior of the memory system all customers "intercept", then the expansion of the ACM module into built into the VMM. According to the constitution of the subject and the object, operation, operation authority and other elements of the ACM security policy table, safety strategies analysis of the behavior, and the corresponding operation according to the matching strategy after the results of (permission, prohibition, alarm). In the isolated system, main body refers to a security level information customer OS object is provided by the Domain0 to the customer OS for data exchange between a shared memory area, the same information with the writer's information security level. The ACM extension module according to the access control strategy table matching strategy. Access control configuration management strategy for the table itself is responsible for the operation of control by software in Domain0 management. Finally, through

the security isolation to control Grant Table memory between virtual machines.

## III. Experimental Validation and Result Analysis

### A. Experimental Verification

By attempting to illegally mapping DomainU virtual machine memory in Domain0, to test the DomainU memory data are protected, design of memory attack experimental steps are as follows:

a) Virtual machines running vm01, in a virtual machine (XP SP3) in the running process of notepad.exe.
b) The second step: in Domain0 try to get DomainU running in process.

Fig. 3 is the memory safety isolation before memory attack experimental results. If the operating system internal memory is safe, then the outside of virtual machine is unable to obtain the memory data of internal virtual machine own space. But the attack by experimental results proved that any personnel as the administrator or to access the Domain0 can access the user space memory (notepad.exe).

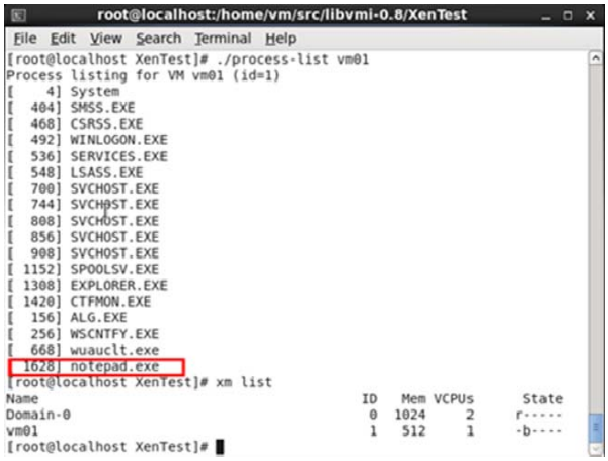


Fig. 3 Design framework

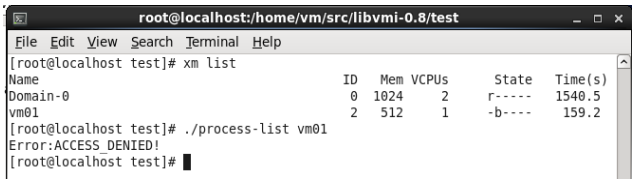


Fig. 4 Memory isolation test result

As shown in Fig. 4 is to take the memory safety isolation after attack experimental results. Because of the existence of memory isolation module, Domain0 cannot read DomainU memory, DomainU memory access list failed, error code Access_Denied.

### B. Result Analysis

The process of acquiring list is a typical Domain0 to read and analyze DomainU memory operation, we use process-list prepared by the procedure successfully acquired the process virtual machine list for in attacking the tests. Join the isolation system, when the Domain0 to access the DomainU memory, it first tries to access the shared memory, and then will need to exchange information written to the shared memory, the information security level are also written into the shared memory. The procedure can be HOOKS module in Xen to be intercepted by the super call interception mode, and the information exchange level mark and its use of the shared memory descriptor (Grant Reference) stored in the Xen internal information exchange access control module (ACM), completed by incident notification Event Channel mechanism of DomainU. DomainU after receiving the notice, try to obtain the shared memory descriptor corresponding, and submit the memory read request by Domain0, while the HOOKS mechanism in Xen to obtain the requested operation, and the operation request containing information (descriptor information level, access information) submitted to the ACM module. ACM module according to the ruling access control policy, if meet the access request that will be through, otherwise reject and record alarm log.

The above proposed method based on Xen memory safety isolation between different domains of memory access in the process of implementation of the security isolation of memory data, to achieve the desired results.

## IV. Conclusion

The cloud environment security has become the current increasingly hot topic, cloud development redefined the concept of security in computer industry, based on the traditional security and cloud security of the transition period, people tend to neglect some safety problems. The study is on the cloud environment security as the breakthrough point, in the Xen environment does not interfere with the normal operation system, and do isolated data between Domain0 and DomainU at the same time. Finally completed the memory safety isolation scheme is proposed in this paper and has verified the validity and index of isolation, and has achieved the anticipated target.

With the rapid development of Xen and cloud related technology today, Xen for security needs far more than mentioned in these. In this paper, based on the current status of Xen used, provides a more complete memory protection scheme for it. Although the program can to improve the safety in the current environment, but there is no absolute security, need to step by step on the next step of research.

## Acknowledgment

The authors thank the lab team for the support of the research on memory safety isolation system.

## References

[1] Barham P, Dragovic B，Farser K, et al, Xen and the Art of Virtualization (J). ACM, 2003(08):164-177
[2] Intel. Intel Virtualization Technology: Hardware support for efficient processor virtualization(s) Intel
[3] Liangliang Huang, Jun Han, Lunwei Wang. Research on (J). security communication mechanism of Xen hardware virtual machine based on computer security,2010(03): 30-46.

[4] Dengguo Feng. Open the security of Cloud Computing Era(J) Information network security, 2011(03): 1-2
[5] Xiaoqiong Guo. Research on Xen virtual machine and memory management (d). Shanghai: Shanghai Jiao Tong University, 2008, 50-63
[6] Reiner Sailer, Trent Jaeger, Enriquillo Valdez, et al, Building a MAC-Based Security Architecture for the XEN Open-Source Hypervisor(c)//21st Annual Computer Security Applications Conference, December 2005, Arizona, 2005.
[7] David C. The Definitive Guide to the Xen Hypervisor (M). Prentice Hall, 2008
[8] University of Cambridge. XEN Interface Manual(M).UK, University of Cambridge，2006:19-20.
[9] Avi Kiviy, Yaniv Kamay, Dor Laor, et al. VM: The linux virtual machine monitor (C) proceedings of the Linux Symposium, Canada, June 2007.