AN ACCESS CONTROL MODEL OF VIRTUAL MACHINE SECURITY

QIN Zhong-yuan[1,3], CHEN Qi[2], LV You[2], QIANG Yong[2], GUO Ai-wen[2], SHEN Ri-sheng[1,3], Zhang Qunfang[4]

[1]Information Science and Engineering School,
Southeast University, Nanjing 210096, China
[2]Wu Jainxiong Department,
Southeast University, Nanjing, Jiangsu 210096, China
[3]Key Lab of Information Network Security,
Ministry of Public Security, Shanghai 201204, China
[4]Computer Department,
Nanjing Institute of Artillery Corps , Nanjing, Jiangsu, China

ABSTRACT

Virtualization technology becomes a hot IT technology with the popu-larity of Cloud Computing. However, new security issues arise with it. Specifically, the resources sharing and data communication in virtual machines are most concerned. In this paper an access control model is proposed which combines the Chinese Wall and BLP model. BLP multi-level security model is introduced with corresponding improvement based on PCW (Prioritized Chinese Wall) security model. This model can be used to safely control the resources and event behaviors in virtual machines. Experimental results show its effectiveness and safety.

1. INTRODUCTION

As one of the key technologies of cloud computing, virtualization has been widely used, this makes the security problem of virtual machines (VM) more and more prominent [1]. Compared to traditional computer systems, virtual machines will meet more security threats; the threats come not only from outside, but also from inside, which are caused by the characteristics of the virtual machine system [2-4]. Nowadays the internal threats increasingly aroused academia and industry interests. The main internal threats include communication attacks between virtual machines, denial of service and information leakage, etc.

Communication attack is a kind of attacks that attackers implant code and attack through the process, shared memory or memory error, which is also called covert channels. There is no satisfied solution for this problem in the industry yet [5-7]. Denial of service makes the computer resources cannot be allocated to the target machine if the malicious VM apply for resources unlimitedly.

Sundarapandian et al. (Eds) : ACITY, AIAA, CNSA, DPPR, NeCoM, WeST, DMS, P2PTM, VLSI - 2013
pp. 133–145, 2013. © CS & IT-CSCP 2013 DOI : 10.5121/csit.2013.3414

The information leakage can happen in the following two situations: (1) The attackers transmit data through covert channels, such as using the time delay between cache miss detection in the CPUs which share the same cache, such attacks usually have some alliance relations, they achieve key data transmission by using the system holes through consultations. (2) The attackers use the resource shared between different VMs, such as shared memory or CPU, to steal information. This kind of data leakage will not cause damage to the system or data to the virtual machine system, but if the VM running in upper layer is a server which contains critical data, the leakage of such information can bring serious risks.

Usually the operation of the virtual machine is controlled by the Virtual Machine Monitor (VMM, also known as Hypervisor). Secure hypervisor can realize the resource isolation, data security, communication security and integrity of their code. Representative work is the IBM's hypervisor architecture: sHype [8]. It uses security model to control system processes, memory access and isolation of internal resources through access control module (ACM). But when the Chinese Wall policy is adopted to prevent the hidden stream passing through different virtual machines, sHype statically divide the different conflict resources into different access area, as a result the resource utilization is reduced [9]. Cheng et al proposed a Chinese Wall model with a priority (Prioritized Chinese Wall, PCW) [10] to reduce the risk of the hidden information flow in a virtual machine system. Based on the previous work, especially sHype and PCW architecture, we propose an access control model suitable to virtual machine environment, and we also verify the model in Xen.

The main contributions of this paper are:

1. An access control security model of virtual machines is proposed, it introduces the BLP(Bell-lapadula) model on the basis of the PCW model to control the memory sharing among virtual machines and maintain the CW model to manage the virtual machine start.
2. The BLP model is improved, the control range of trusted subjects and security level are added.
3. We implement a prototype for the security model, three strategies are used to realize the security model. Results show the effectiveness of the proposed model.

2. MODEL ANALYSIS

2.1 Analysis of virtual machine access control model

In 2005 IBM proposed sHype, which is a security architecture used to control the information flow between the operating systems that share the same hardware platform and it can achieve the controlled sharing of resources between virtual machines. Xen virtual machines adopted the core idea of sHype that it set an ACM module in the virtual layer and implemented two security policies [11]: Chinese Wall (CW) [12] and Simple Type Enforcement (STE). Its main objectives are: (1) Management of the communication between two virtual machines in the same system; (2) Management of the hardware resources that virtual machine can access; (3) Multiple virtual machines in the same conflict set of interest cannot be run at the same time, this reduced the occurrence of the convert channel. While the CW strategy can prevent virtual machines have conflict of interest running at the same time in the same VMM, the STE policies can control whether communications can happen between different virtual machines and the hardware resources virtual machines can access.

CW and STE both have significant advantages on the control and distribution of resources, which is the main reason why Mandatory Access Control (MAC) method can be well applied in VMs. However, CW and STE also have many limitations, which mainly reflected in that they cannot prevent the covert channel; an attacker can easily take advantage of shared resources or third-party virtual machine for data transfer. Ge Cheng, Hai Jin et al proposed the PCW strategy model [10], which is based on the Chinese Wall security policy, but it is also different with the sHype's CW strategy, i.e., the relationship of conflict sets is no longer static, but can be extented along with the flow of information. This model can prevent the convert channel effectively, however, the following defects also existed:

1. PCW model is so strict that it greatly reduced the resource utilization of virtual machine, which is contrary to the original goal of virtual machine design, and there is also a possibility of system deadlock that a virtual machine may never get a chance to run because of dynamic conflict set of interest unless you restart the system.

2. PCW is only limited to isolated control of virtual machine resource allocation, that is, either access or refused, but this does not match the actual requirements. Virtual machine is equivalent to a computer group, there are two conflicts between the virtual machines in this group, one is the resource request and the other is access rights. But restrictions on access rights cannot be strictly isolated. For example, FTP service, when there is a conflict between the host and client access rights, the host can set the access rights of the client, limit the client's read and write permissions so that client can operate within its scope, but PCW can only permit or deny which cannot fit this demand.

2.2 BLP model and its improvement

From above, we can see that the CW and PCW conflict sets cannot fully meet the security needs of the virtual machine. In order to solve the problem from the security of the whole system, we introduce another model: BLP multi-level security model. In our model, BLP will be used to control the memory sharing of virtual machine, it also manage the CW model to control the start of virtual machine. Considering the PCW can effectively prevent the covert channel threats brought by the communication and CPU resource sharing of virtual machine. Therefore, CW, PCW and BLP will simultaneously exist in our model. We have made some improvements on the BLP to better meet the system requirements.

BLP model was proposed for the U.S. military in the 1970s to solve the information security and confidentiality problems of time-division system, the model is mainly used to prevent confidential information to be accessed by unauthorized subject [13]. In BLP model, set $L(s) = l_s$ as security permissions of subject s, set $l(o) = l_o$ as sensitivity level of object o and for all the security levels l_i $(i = 0,1,\ldots\ldots,k-1, l_i < l_{i+1})$, there are three access and control rules:

1. Independent safety features: the subject s has discretionary access permissions on the object o.
2. Simple safety conditions: the subject s can perform read operation on object o, if and only if $l_o < l_s$, s has autonomous read permissions on o.

3. *-properties: subject s can perform write operation on object o, if and only if $l_o > l_s$, s have autonomous read permissions on o.

Considering the special features of VMs, we made the following improvements on BLP:

1. The trusted subject is added

BLP model's strict confidentiality brings the system a lot of restrictions, e.g., any virtual machine can communicate only based on the security level, however, some special virtual machines need to exchange information unconditionally. Therefore, we add a trusted subject, which can go beyond the BLP model's strict security level system and exchange information with the target object. We make a limitation that trusted subject cannot exchange malicious objects with other objects, i.e., a trusted subject cannot be served as malicious object transfer medium.

2. Control of the security level's range

In the conventional implementation of BLP axiom, a subject's sensitive level is fixed in its entire life cycle. The system set a security level for each VM, but it does not mean that all the virtual machine of higher security level have read permission on those of lower security level, because it does not meet the actual service's demand. Therefore, there is a need to set a zone classification for the security level. There are multi-level security data streams in a security zone, while it does not exist in the different zones. In this way the flow of data between the virtual machines in different security levels can be controlled according to system requirements.

3. MODEL DESIGN AND SECURITY ANALYSIS

3.1 Model Design

In our model we assume that the security levels of all the virtual machines are in the same zone, meanwhile virtual machines have independent access rights to the resource allocated by the system, that is, the subject and object both meet the discretionary access permissions. We present the Virtual machine Based Access Control model (VBAC), which is defined as follows:

1. Model elements

- Subject and object elements

Subject set $S:\{s_1, s_2, \ldots\ldots, s_m\}$, defined as an object which sends access request, **Object set** $O:\{o_1, o_2, \ldots\ldots, o_m\}$, defined as an object is accessed. In the virtual machine system, both VM and system resources are likely to become one of subject or objects;

Trusted subject set : $T(s):\{s \mid s \text{ is trusted and } s \in S\}$;

- Access characteristic and request decision sequence

Access characteristic (AC) contains memory read and write: $\{read, read \mid write\}$, memory transfer: $\{mem-tranfer\}$, VM label**:** $\{addlabel, rmlabel\}$, resource apply and release: $\{apply, release\}$, VM create and destroy: $\{create, destroy\}$, VM start and stop: $\{start, stop\}$, communication apply and

release: $\{com-apply, com-release\}$, **security level adjustment**: $\{level(L_u) \mid L_u \text{ is the new security level}\}$;

The state set of the virtual machine : State(o){running, stop, sleep};

System security level F: security category, f1 and f2 represent subject's and object's security level respectively, f3 and f4 represent category;

Request set $R: S^+ \times O \times X$, $S^+ = S \cup \{\phi\}$, $X = AC \cup \{\phi\} \cup F$, request element is $R_i: \{R_i \mid R_i \in R, i \geq 0\}$;

Security Decision set *D:{yes,no,error,?}*, its element is $D_j: \{D_j \mid D_j \in D, j \geq 0\}$, which is the decision of request R_j;

Request sequence $X: R^T$, which represents the request sequence at different time, request element is x, x_t represent the request at time t;

Decision sequence $Y: D^T$, it's the decision sequence set which security policy responses to the request sequence, decision element is denoted as y, y_t represents the decision at time t;

- The conflicts set and access matrix

The conflicts interest set: *CIS(o)*, the data set which has interest conflict with object *o*, element $CIS_t(o)$ represents the conflict relation set of *o* at time t; conflict zone: RC, all the elements in a conflict set belong to the corresponding conflict zone, that is $\forall s \in CIS(s), s \in RC$; if s is not included in any conflict set, $\neg \exists s` \in S, s \in CIS(s`)$, it's the same to say s belongs to no conflict zone, then define $s \notin RC$;

System access matrix *A: (S×O)*, it's used to record the access memory of the subjects and objects controlled by the system, matrix element $A_t(s_i, o_j)$ represents the access matrix at time t, $A_t \to \{-1, 0, 1\}$,1 represents that s_i had accessed o_j, 0 represents that it's still not decided and will change with decision $D_t(s_i, o_j, x)$, -1 represents that it cannot be accessed .A_0 is the initial access matrix.

- System state and system

  Present access set: $P(S \times O \times AC)$;

  System state V: $P(S \times O \times AC) \times A \times F$;

State sequence $Z: V^T$ is the state sequence, state element is denoted as z, z_t represent the state of time t; z_0 is the initial state, usually regard initial state of system as safe.

State shift: $W \subseteq R \times D \times V \times V$ represent a state shift to another state after request and decide;

$SYM : \sum (R, D, W, z_0) \subseteq X \times Y \times Z$, $(x, y, z) \in SYM$, only at any time $t \in T$ $(x_t, y_t, z_t, z_{t-1}) \in W$;

2. Model rules

To implement the security control of the system, the subjects and objects of the virtual machine system is defined as the safety rules below. In B1, B4, B5, B6, B7, B8, B9, and B10, subjects and objects are VMs, but in other rules the subjects are virtual machine and the objects are the hardware resources.

Rule B1: The subject add a label or remove a label , $R_t(s, o, addlabel) \mid R_t(s, o, rmlabel)$

$R_t(s, o, addlabel) \mid R_t(s, o, rmlabel)$ is permitted only when $s \in T(s)$, that is, the subject is a trusted subject.

Rule B2: The subjects apply for resource, $R_t(s, o, apply)$

$R_t(s, o, apply)$ is permitted only when the following condition is satisfied.

1. $A_t(s, o) \neq 1$, s had never accessed o before t;
2. There's no other subject is accessing o at t;
3. $\forall s' \neq s : A_t(s', o) = 1 \wedge \forall o' \in CIS_{t+1}(o) : A_t(s', o') \neq 1$, the subjects had accessed o will not be conflicted with s .

Rule B3: Subjects release resources, $R_t(s, o, release)$

$R_t(s, o, release)$ is permitted only when $state(s) = stop$, that is, the subject is stopped.

Rule B4: Subjects create and destroy the objects, $R_t(s, o, create \mid destroy)$

$R_t(s, o, create \mid destroy)$ is permitted only when $s \in T(s)$;

Rule B5: Subjects start the objects, $R_t(s, o, start)$

$R_t(s, o, start)$ is permitted only when: $s \in T(s)$ and $\forall o$`, $Dt(o, o`, apply) = 1$, that is, all the resources applied by the client is permitted by Rule 2;
$\forall s' \mid (state(s') = running) \wedge (state(s') = sleep), o \notin CIS(s')$, the object is not conflicted with the VMs which is running or suspended.

Rule B6 : The subjects stop the objects, $R_t(s, o, stop)$

$R_t(s, o, stop)$ is permitted only when $s \in T(s)$.

Rule B7: The subjects communicate with the objects, $R_t(s,o,com-apply)$

$R_t(s,o,com-apply)$ is permitted only when:

1. $s \notin RC \| o \notin RC$, that is, s and o don't belongs to any conflict set;
2. $s \notin CIS(o) \| o \notin CIS(s)$, that is, the two sides of communication aren't in the same conflict set.

Rule B8: The subjects release communication, $R_t(s,o,com-release)$

$R_t(s,o,com-release)$ is permitted only when $D_t(s,o,com-apply)$=1, that is, if the communicating rules between s and o fit B7 and get permission, release communication is always permitted.

Rule B9: The subjects adjust the security level, $R_t(s,o,level(L_n))$

$R_t(s,o,level(L_n))$ is permitted only when $s \in T(s) \wedge (state(o) = stop)$,that is, s is trusted subjects and the client is not running.

Rule B10: The subjects apply for memory transfer from the objects, $R_t(s,o,mem-tranfer)$

$R_t(s,o,mem-tranfer)$ is permitted only when $(f_1(s) > f_2(o) \wedge f_3(s) \supseteq f_4(o)) \| (s \in T(s))$; that is, the subjects dominate the object or the subject is trusted subject.

Rule B11: The subjects only read the mapped memory, $R_t(s,o,readonly-map)$

$R_t(s,o,readonly-map)$ is permitted only when: $(f_1(s) > f_2(o) \wedge f_3(s) \supseteq f_4(o)) \| (s \in T(s))$, that is, the subjects dominate the object or the subject is trusted subject.

Rule B12: The subjects read or write the mapped memory, $R_t(s,o,read \mid write)$

$R_t(s,o,read \mid write)$ is permitted only when:

$\{(f_1(s) = f_2(o)) \wedge [f_3(s) \supseteq f_4(o) \| f_4(s) \supseteq f_3(o)]\} \| (s \in T(s))$, that is, the two sides of read and write are at the same security level.

Conflict set expansion theorem:

1. If $R_t(s,o,apply)$ fit B2, then:

   a) If $(\forall s_1 \in SYM, s \in CIS_t(s_1)) \wedge (\exists s_2 \in SYM, A(s_2,o) = 1)$, let $A(s,o) = 1$, expand the conflict set of s_2 as:

$$CIS(s_2) = \bigcup_{s_2 \neq s_1 | A(s_2,o)=1} s_1$$

   b) If $(\forall s_1 \in SYM, s \notin CIS_t(s_1)) \wedge (\neg \exists s_2 \in SYM, A(s_2,o) = 1)$, then only let $A(s,o) = 1$,that is, if s is not in any conflict set and o was never accessed by any object, we only need to modify the access history matrix A;

(2) If $R_i(s,o,com-apply)$ fits B7, then:

a) If $s \notin RC \| o \notin RC$ (s or o doesn't belong to any conflict set, then expand the conflict set:

$$\left\{ \begin{array}{c} CIS(o) = \bigcup_{s \neq o} s, s \notin RC \wedge o \in CIS(o) \\ CIS(s) = \bigcup_{o \neq s} o, o \notin RC \wedge s \in CIS(s) \\ do\_nothing, o \notin RC \wedge s \notin RC \end{array} \right\}$$

b) $s \notin CIS(o) \| o \notin CIS(s)$, if the two sides of communication aren't in a same conflict set, expand any of them:

$$CIS(s) = \bigcup_{s \neq o} CIS(o)$$

So,

$\exists (s_1, s_2, \ldots, s_N), (D(s_1, s_2, com-apply) = 1) \wedge (D(s_2, s_3, com-apply) = 1) \wedge (\ldots\ldots) \wedge (D(s_{N-1}, s_N, com-apply) = 1)$, will expand all conflict sets of $s_i (1 \le i \le N)$:

$$CIS(s_1) = \bigcup_{s_i \neq s_1}^{2 \le i \le N} s_i$$

By symmetry and transitivity, we can be obtained that all the subjects of direct or indirect communication will in a same conflict set, which is the previously mentioned Alliance of conflict set.

3.2 VBAC model security analysis

In VBAC model, rule B1 to B8 of VBAC adopt the core ideas of CW's conflict set; B9 to B12 adopt the multi-level security idea of BLP.

We assume that the initial system state is safe in a conflict of interest and secure confidentiality. Rule B2, B3 control resource allocation, in full compliance with the idea of conflict isolation, meet CW's simple security features to ensure there's no information disclosure on the resource allocation. Rule B7, B8 control event channels between virtual machines by CW's idea, event channel is a technical core of the virtual machine, as semaphore mechanism of the traditional system. Many virtual machine behaviors are based on the event channel, if the control is too strict, the availability of the virtual machine will be affected, so the control of event channel need to be set in accordance with the secure requirements of the system. There are a third party covert channel in Rule B2, B3, B7, B8, the conflict set expansion theorem is proposed to solve this problem and eliminates the third-party security risks well.

The model rules B9 B12 control the virtual machine resource sharing and communication problems by idea of BLP, rule B10, B11, B12 meet the security features of the multi-level security system, thus, the system conversion will limited by B9 to B12 to ensure the confidentiality.

From the model, we can better avoid information leakage caused by system resource allocation, as well as better control of third-party covert channel of virtual machines.

4. IMPLEMENTATION AND EXPERIMENT RESULTS

In the implement, we divide the virtual machine-based access control model into three parts: the main strategy, the second strategy, the third strategy. The secure control strategy is realized through the cooperation of three strategies.

(1) The main strategy primary policy: It implemented the conflict set model, used core idea of CW and is responsible for create, start and destroy of the virtual machine. The strategy is based on virtual machine and follows VBAC model's Rule B1, B4, B5, B6;

(2) The second strategy second policy: In Xen, resource control and event channel management uses STE model, we also use this security model. STE adopted CW's idea of interest conflict set, it is responsible for application, allocation, revocation of virtual hardware resources, and the event channel management, in the model proposed in this paper, it will follow the VBAC model's rule of B1, B2, B3, B7, B8, as well as the conflict set expansion theorem;

(3) The third strategy third policy: The third strategy controls the multi-level secure communications between virtual machines; it adopted the core idea of the BLP, and is responsible for managing memory sharing, data transfer and other issues. It follows the VBAC model rule B9, B10, B11, B12.

4.1 Test Results

1. Memory management test

In our experiment the PC's physical memory is 4GB, that is, number of pages in the memory space: 4G/4KB = 1M. Test scenarios of memory access control are as follows: create three VMs Dom1, Dom2, Dom3, configuration information for each VM memory is: memory 512MB for Dom1, memory 256MB for Dom2, and 256MB for Dom3.

(1) Scheme 1: test if there is conflict relationship between Dom1 and Dom2;

These two virtual machines are allocated memory space respectively. The test procedure is as follows: ① Both Dom1 and Dom2 were not set a security type, after starting Dom1 detect the memory space allocated by Xen; ②close Dom1, then start Dom2 and detect memory space allocated by Xen.

We get memory allocation of Dom1 and Dom2 through Xentrace tools and debug-keys in Xm as shown in Figure 1. In figure 1, the vertical axis represents the time axis, the horizontal axis represents the number of pages, and experimental machine has 4GB physical memory or memory pages 1M, so the maximum memory page number is 1048576. In the Xen virtual machine system, Xen occupied the starting 64MB memory space, as the red region shown in the figure; after start of the Domain0, all the other memory was managed by Domain0 and it got a memory allocation of 131,072, as the green region shown; When start Dom1, it was allocated a memory of 65536

pages, its page space is the yellow region shown in Figure1, after closing Dom1, Dom1's memory page was recovered; so when starting Dom2, Domain0 allocate memory for it in-memory heap, as the black region shown in the figure. We can conclude from the data that Dom2 and Dom1 can share memory pages.

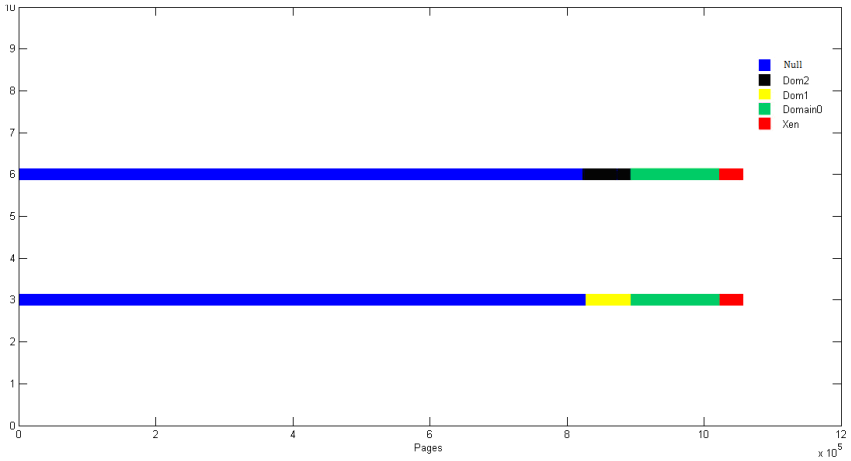

Fig. 1. Control of Memory resource test results

(2) Scheme 2:

Test the memory space which is allocated for the virtual machine of same conflict set.

The test procedure is as follows:Label the Dom1 with security type "A", and label the Dom2 with "B", after switch on the Dom1, test the memory address segment assigned to Xen; ② Close Dom1, switch on Dom2 ,examine the memory page status allocated for Dom2 . Figure 2 shows the results of scheme 2. In the design of scheme, there are some conflict of interests between Dom1, and Dom2.

Thus, despite Domain0 recover its memory after Dom1 is closed, the type of access has been recorded in memory by the history array.

The memory page used by Dom1 cannot be assigned to Dom2 when allocating memory pages for Dom2 because the conflict relationship mentioned above. The memory page used by Dom1 is marked in yellow, page of memory allocated for Dom2 is marked in bright blue, from the statistics, we find that there is no shared memory page in Dom1 and Dom2.

(3) Scheme 3:

Test the influence that conflict set expansion theorem may have on memory allocation. In order to achieve the expected results, we set memory configuration of Dom1 to 512MB, the memory of Dom2 and Dom3 to 256MB.

The test procedure is as follows: Set all the array which contain memory usage history to 0, the Dom1 with security label "A" , Dom2 with security label "C", examine the memory address ranges assigned to Xen after switched on Dom1; ② Close Dom1 and switch on Dom2, test status of memory page allocated for Dom2; ③ create Dom3 which is set with security label "E" , then close Dom2 and switch Dom , test status of memory page allocated for Dom3.

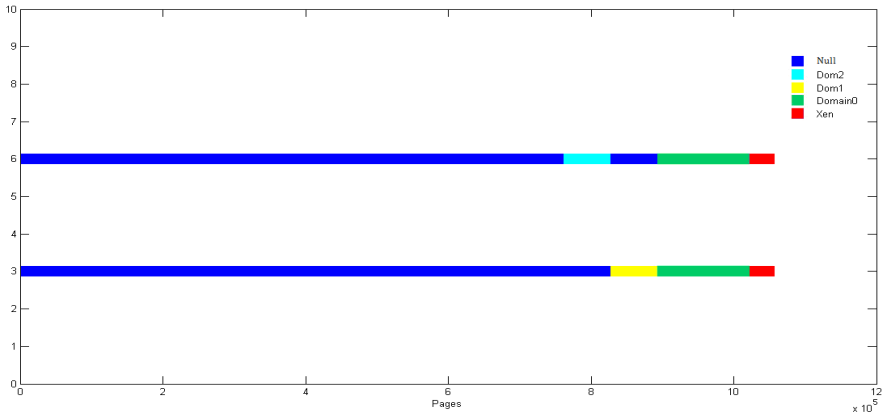

Fig. 2. Result of memory resources control test in scheme 2

Three memory allocation of the test scheme 3 is shown in Figure 3. 131,072 memory pages used by Dom1 are marked in yellow, Dom2 represent memory pages for Dom1 which is allocated by Domain0 after Dom1 is closed; in the scheme, there are some conflict between Dom2 and Dom3, for they had shared the same memory page. According to conflict set expansion theorem, they had the same type of conflict. Thus memory page used by Dom1 cannot be assigned to Dom3, In Figure3 memory allocated for Dom3 is marked in bright blue.

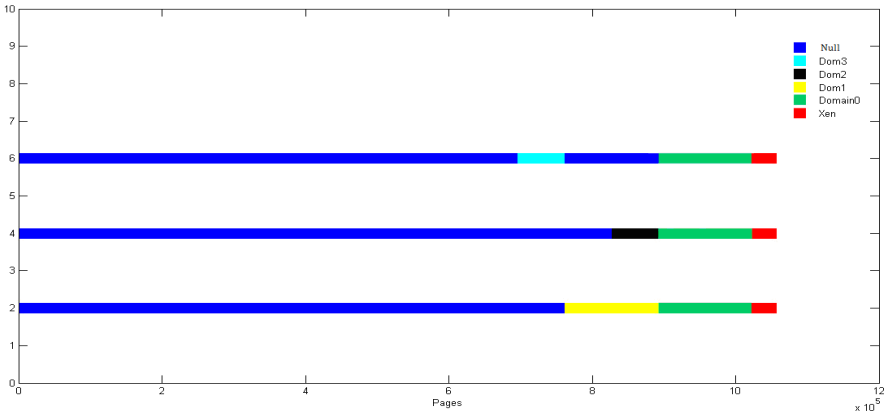

Fig. 3. Result of memory resources control test in scheme 3

4.2 Result analysis

From the test data and result analysis, we find the proposed security model has the following several characteristics:

(1) High safety.

In the section about security and effectiveness in this paper, the safety of virtual machine management, event channel, resource management and shared memory access control have been tested. Results show that the virtual machine management control accord with the requirement of CW set model of interest conflict; Resource management and event channel control can get the effective control through extended definition of conflict set; the control to memory shared follows multistage safety control rules of VBAC model which has been mentioned in this paper.

(2) Less space overhead.

In access control module, the security level and conflict set type stored in arrays after the mapping, the size of strategy information cache array is decided by security type number M and virtual machine number N. Assuming that each array element occupied 1 Byte, then the array of strategy information will occupy M * N Bytes totally. In resource controlling, the largest overhead comes from the memory pages. In order to reduce the cost of the space, we adopt the Bit-map storage, which has greatly reduced the space overhead.

(3) Existence of time performance loss.

From the testing process, we can see that performance loss in access control are mainly caused by memory allocation, because a lot of memory page need to be allocated every time when the system starts, and each allocation will cause reading and decision of model information, thus a certain period of time loss happened.

5. SUMMARY

Virtual machine is one of the key technologies of cloud computing, but there exist many attacks to the virtual machine. According to the internal safety problems of virtual machine, we propose a novel access control model which is suitable for virtual machine environment. On the basis of PCW security model, we introduced the BLP multilevel security model, and make corresponding improvements of BLP. The performance and space overhead are ananlyzed. The simulation results show that the proposed model is feasible and secure.

ACKNOWLEDGMENTS

This work is supported by the Key Lab of Information Network Security, the Ministry of Public Security and Information Security Special fund of National Development and Reform Commission (Project name: Development of Security test service capabilities in wireless intelligent terminals).

REFERENCES

[1] Feng DG, Zhang M, Zhang Y, Xu Z. Study on cloud computing security[J]. Journal of Software, 2011,22(1):71-83.
[2] Wang, Z.,X.X. Jiang. HyperSafe: A Lightweight Approach to Provide Lifetime Hyper-visor Control-Flow Integrity[C]. 2010 IEEE Symposium on Security and Privacy, 2010:380-395.
[3] Salaun, M. Practical overview of a Xen covert channel[J]. Journal in Computer Virology, 2010, 6(4): 317-328.
[4] Liu, Q., G.H. Wang, C.L. Weng, et al. A Mandatory Access Control Framework in Vir-tual Machine System with Respect to Multi-level Security II: Implementation[J]. China Communications, 2011, 8(2): 86-94.
[5] Ranjith, P., C. Priya,K. Shalini. On covert channels between virtual machines[J]. Journal in Computer Virology, 2012, 8(3): 85-97.
[6] Okamura, K.,Y. Oyama. Load-based covert channels between Xen virtual machines[C]. 25th Annual ACM Symposium on Applied Computing, Sierre, Switzerland, 2010:173-180.

[7] JingZheng, W., D. Liping, W. Yongji, et al. Identification and Evaluation of Sharing Memory Covert Timing Channel in Xen Virtual Machines[C]. 2011 IEEE 4th Interna-tional Conference on Cloud Computing (CLOUD 2011), Los Alamitos, CA, USA, 2011:283-91.

[8] R. Sailer, E.V., T. Jaeger, R. Perez, L. van Doorn, J. L. Griffin, and S. Berger. .sHype: Secure Hypervisor approach to trusted virtualized systems, Yorktown Heights, NY, USA:2005.

[9] CHENG Ge, JIN Hai, ZOU De-qing, ZHAO Feng. Chinese wall model based on dy-namic alliance[J]. Journal on Communications, 2009, 30(11): 93-100.

[10] Cheng, G., H. Jin, D.Q. Zou, et al. A Prioritized Chinese Wall Model for Managing the Covert Information Flows in Virtual Machine Systems[M]. Proceedings of the 9th In-ternational Conference for Young Computer Scientists, Vols 1-5, ed. G.J. Wang, et al.Los Alamitos: Ieee Computer Soc,2008.

[11] Shi Nei, Zou Deqing, Jin hai. Xen Virtualize technology [M]. Wu han: Huazhong Uni-versity of Science and Technology Press,2009.

[12] Foley, S.N. Building Chinese walls in standard unix(TM)[J]. Computers & Security, 1997, 16(6): 551-563.

[13] Gansen, Z.,D.W. Chadwick. On the modeling of Bell-LaPadula security policies using RBAC[C]. 2008 IEEE 17th Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises., Piscataway, NJ, USA, 2008:257-62.