The Design and Implementation of Security Defense Technology Based on Mandatory Running Control

Liu Jizhen, Wu Kehe
Department of Automatic Control, Department of Computer Science and Technology
North China Electric Power University
Beijing, China
nepuljz@263.net, epuwkh@126.com

Zhang Tong, Ma Gang
Department of Computer Science and Technology
North China Electric Power University
Beijing, China
zhtzhangtong@163.com, hdmagang@163.com

Abstract—In the multi-processed and multi-threaded complex network environment, application system will appear the unforeseen deadlock problem. This deadlock problem may cause service interruption, and even can cause the crash of the application system. On the other hand, malicious attacks on the system and other random events will also cause serious consequences. The security defense of service operating has become another important problem in the information security field. In order to solve the above problem, this article proposes the security defense technology based on Mandatory Running Control, which can solve the unpredictable deadlock problem from the system kernel level, prevent illegal process running and restrict allowed process’s behavior, realizing security defense of service operating and ensuring the system running continuously and reliably. This paper introduces the principle, theoretical model and the technologies in the implementation of the security defense technology based on Mandatory Running Control.

Keywords—Mandatory Running Control; Security Defense; Monitoring Process; Kernel Protection

I. INTRODUCTION

With the development of network and computer technology, computer security has increasingly become a pressing problem. The protection of computer information’s confidentiality, integrity and availability has been the main objectives of the information security study for a long time. But with the development of network, people have begun to recognize that service normally running without interruption and damage is very important in many cases. In complexity network environment, application system will meet the unforeseen deadlock problem, that is the system halts. The deadlock problem can cause the interruption of service and even can cause the crash of application system. On the other hand, malicious attacks on the system and other random events will also cause serious consequences. Because of the above problems, the security defense of service operating has become more and more important. In order to solve the above problems, we propose the security defense technology based on Mandatory Running Control in this paper, which can solve the unpredictable deadlock problem from the operating system kernel level. In the same time, the technology can restrict processes’ behavior through a new monitoring and scheduling process technology to prevent malicious attacks and ensure application system and operating system running normally without interruption and damage.

II. THE BASIC PRINCIPLES OF SECURITY DEFENSE TECHNOLOGY BASED ON MANDATORY RUNNING CONTROL

The security defense technology based on Mandatory Running Control can solve the unpredictable deadlock problem from the operating system kernel level. In the same time, the technology can prevent illegal processes running and restrict allowed processes’ behavior through a new monitoring and scheduling process technology, ensuring application system and operating system running normally without interruption and damage.

Specifically, the security defense technology based on Mandatory Running Control ensures application system’s security from the following two aspects.

1) A new monitoring and scheduling process technology restrict processes’ behavior to prevent malicious attacks: Traditional methods and tools of monitoring and scheduling process are based on API interface or system calls which are provided by operating system. These methods and tools can not get user required information initiatively from the operating system kernel data structure and report the current system status to users real-timely and effectively. On the other hand, the traditional methods run in the user level, so they can easily be replaced or deleted by invaded hackers. All of the above problems can affect the reliability of application system. To solve the above problems, we design a new monitoring and scheduling process technology. By kernel data structure analysis, we extract the data structure which monitoring process program required. The monitoring process program gets the required data
from the operating system kernel files. In the same time, the “allowed process list” which records all safe and reliable processes and the “process access list” which records the permissions of every allowed process restrict the behavior of processes and prevent malicious attacks effectively.

2) To solve the unforeseen deadlock problem from the operating system kernel level, in complex network environment: Using the information got from the kernel files, monitoring program real-timely monitors the key service processes to detect the anomalist processes which occupy excessive system resources and then terminates the anomalist process to release the resources the process occupies. After the prescript time t, the terminated process is resumed. When the service process is terminated, the client process, which requires service from the service process, could not connect with the service process normally. In order to ensure the service integrity, the client process has the re-connected mechanism. When the service process is resumed, the client process can require service normally.

III. THE MODEL OF SECURITY DEFENSE TECHNOLOGY BASED ON MANDATORY RUNNING CONTROL

After introducing the basic principles of the security defense technology based on Mandatory Running Control, now we model the technology as follows:

A. State Set: \( S = \{s, s_0\} \)
   
   \( s \) represents the current operating system state, \( s_0 \) represents the safe and stable running operating system state.

B. Process Set: \( P = \{p_1, p_2, \ldots, p_n, \ldots, p_m\} \)
   
   \( n \) represents the number of system processes.

   For each process \( p_i \), it needs to meet three requirements, which are the availability, integrity and confidentiality. Here we also put forward the confidentiality, integrity and availability three requirements for every process, specifically, as follows:

   Confidentiality: The access behavior of each process \( p_i \) must meet the restrictions in the "process access list". That is, process \( p_i \)'s access to system resources and other processes' access to process \( p_i \)'s data areas are all restricted.

   Integrity: The integrity of the process \( p_i \) is that the content, script, code and data should always keep integral and the service which the process provides must be integral.

   Availability: The current operating system resources occupancy state of process \( p_i \) should not cause the deadlock problem.

   The safe and stable running operating system state \( s_0 \) is that all of the processes must meet the confidentiality, integrity and availability three requirements.

C. Model Constraints

We use the symbol "\( s \equiv s_0 \)" to represent that the current operating system state \( s \) is the same as the safe and stable running operating system state \( s_0 \). In this model, we should determine whether the state \( s \) is the same as the state \( s_0 \). The actions which the state \( s \) uses to transfer to the state \( s_0 \) make up the Action Set \( A \).

\( A = \{\text{none action, protecting the confidentiality action, protecting the integrity action, protecting the availability action}\} \). The following is the course of the transformation between state \( s \) and state \( s_0 \).

If \( s \equiv s_0 \), then the current operating system is in the safe and stable running operating system state and don’t need any action, that is, the none action in set \( A \).

Else, adjust the current operating system state to the safe and stable running operating system state \( s_0 \). The content of adjustment is to find out the processes which are not satisfied with one or more of the confidentiality, integrity and availability three requirements and then perform actions to return the processes to normal state. Specific actions are as follows:

1) protecting the availability action: if the occupancy operating system resources state of a process can cause the deadlock problem, the process will be terminated to release the resources it occupied. After a prescript time \( t \), the process will be recovered to a normally running status. This action can recover the process’s availability, but it also damages the integrity of the process, so that the recovering integrity action should be used at the same time.

2) protecting the integrity action: the protections of integrity need "client reconnection mechanism"; that is, after the service process is killed, its downstream processes will set up communications to the service process again until the service process is resumed. On the other hand, in order to protect the integrity of process’s content, each process is protected.

3) protecting the confidentiality action: the protection of process’s confidentiality is achieved by the "process access list". The list provides the permissions of process’s access to system resources. Process’s access to system resources and other processes’ access to the process \( p_i \)'s data area both must meet the permissions provided by the list. Once the requirements don’t meet the permissions provided by the list, we think the confidentiality of the process threatened and then reject the illegal requirements.
After the adjustment under the above three type of action, we determine whether the state \( s \) is the same as \( s_0 \), then repeat the above operation. The whole course is shown in Figure 1.

IV. IMPLEMENTATION TECHNOLOGIES OF SECURITY DEFENSE TECHNOLOGY BASED ON MANDATORY RUNNING CONTROL

A. Scheduling Processes based on kernel

The security defense technology based on Mandatory Running Control schedules and monitors processes through the "allowed process list" and "process access list".

In the safe system environment, we collect safe processes’ information as comprehensively as possible to develop the "allowed process list", which is the basis for monitoring and scheduling processes\(^7\). In the scheduling procedure, if the current process doesn’t register in the "allowed process list", the program will inform user through the terminal immediately. Only when user confirms by his local keyboard or mouse, the process can be allowed to run. Then the process registers to the "allowed process list". As a result, we prevent the illegal processes running in the system.

At the same time, "process access list" records all of allowed processes’ permissions in the system to restrict allowed processes’ behavior. "process access list" and "allowed process list" ensure that the processes running in the system are all credible and each credible process’s behavior is limited, preventing malicious attacks effectively. The schedule procedure is shown in Figure 2.

B. Monitoring Processes Real-Timely and Effectively

The security defense technology based on Mandatory Running Control monitors\(^8\) all the key service processes in the system real-timely to detect anomalous processes and deal with them. By the statistics and measurement methods, we calculate a nominal value of system resources occupancy for all key service processes in the normal situation. The nominal value works as a threshold. When a process’s occupancy of CPU or memory is more than the threshold, we think the process is anomalous and terminate it to release system resources which the process occupied. After the prescript time \( t \), the monitoring program will restart the process.

Monitoring procedure is shown in Figure 3.

In order to ensure the integrity of the service, the client which request service for service process has re-connected mechanism in the C/S mode. That is, when a service process is terminated by monitoring program, the client process couldn’t connect with the service process, and then the client process will re-connect with the service process until the service process restarts.

C. Kernel Data Structure

In order to get the information of processes from the system kernel\(^6\), we design the following data structure.

```c
typedef struct
{
    char Name[20]; // process name
    _u32 PID;      // process ID
    _u32 PPID;     // father process ID
    _u32 UID;      // process user ID
    _u32 GID;      // process group ID
    char StartTime[20]; // process beginning time
    char Status[20]; // process running status
    double Standard[2]; // process’s occupancy threshold of system resources
    double Cpu;      // the rate of process’s CPU usage
    double Mem;     // the rate of process’s memory usage
} process_info;
```

V. CONCLUSION

The security defense technology based on Mandatory Running Control can solve the unforeseen deadlock problem from the operating system kernel level in complex network environment. At the same time, the technology can prevent illegal processes running and restrict the allowed processes’ behavior, preventing malicious attacks effectively. As a result, the application system and operating system can run normally without interruption, which ensures the security of system in the kernel level.

REFERENCES

Figure 1: The diagrammatic presentation of Mandatory Run Control Model.

Figure 2: The diagrammatic presentation of Scheduling Processes Procedure.

Figure 3: The diagrammatic presentation of Monitoring Processes Procedure.