Design and Implementation of A Security PVM*

Xue Lan Song Jie Chen Ximing

(Dept. of Computer Science, UEST of ChinaChengdu 610054)

Abstract Security issue is the major factor essential to the wide spread of PVM. A proposal is provided to enhance PVM security property. It solves the security problems of interception, interference, forgery and replay, which endanger the messages delivered among computers in a cluster. It is independent of specific protocols of the lower levels of networks, consuming a limited number of system resources and available to dynamic configuration.

Key words parallel virtual machine; cluster computing; parallel computing; networks security; message; interception; interference; forgery; replay

PVM(parallel virtual machine) is the most widely used programming platform applied in parallel cluster system. It exhibits a good performance in portability, efficiency, compatibility for heterogeneous software and hardware system. Based on the concept of virtual machine, computers that consist PVM synchronize computing tasks by means of sending and receiving messages. However, sent in the form of plain text, each message dilivered on the net is facing the danger of being intercepted, interfered and replay. In this case, PVM security issue becomes prominent, which seriously checks the arrangement of PVM in large scale and geographically wide spanned networks, as well as its application in key tasks[1].

The main purpose of PVM is to provide an environment for network parallel computing programming based on message delivering, which can sustain different kinds of software and hardware platforms. Thus, message delivering mechanism of PVM is: 1) of short delay; 2) independent of the lower level protocols.

1 Working Mechanism and Security Fault of PVM

1.1 Working Mechanism of PVM

Virtual machine is the fundimental concept of PVM[2]. By means of integrating heterogeneous computers on the net to one parallel virtual machine in the eye of users, PVM provides a parallel programming environment based on message delivery. Application programs synchronize computing tasks through message delivery, while PVM is responsible for delivering those messages safely and reliably, as well as automaticly transferring data types among different architeture computers.

In the view of software, PVM is a software system consists of two parts: 1) a daemon process running backstage called pvmd, which is active on every computer of the virtual machine, responsible for message delivery and routine of the whole PVM system and everyday management; 2) a running-time library, which is linked to each PVM application program by a linker. When sending messages to other PVM process, the first step for PVM application program is to call message sending primitive in the running-time library, which communicates with the daemon processs at the backstage, then it is up to the daemon process to make the necessary transferring of format to send messages to the destinate.

The definition of message sending primitives are independent of specific networks protocols. They can be complemented either by TCP/IP, or the light-weight protocols.

Received December 6,1999
* The project supported by Information Industry Ministry Development Funds

1.2 Security Fault of PVM

Task identifier is a key concept of PVM, a TID specifies a task on virtual machine[3]. Each TID consists of two parts: host identifier and task identifier. Every task on virtual machine has its own specific TID, which must be included in all of the messages sent to other computers. So that when received, messages can be handled according to the TID. What is worth of menitioning is that TID and messages containing TID are all sent in plain text.

Several security faults exit in the above message delivery mode:

1) PVM messages are delivered by the lower level protocols of the networks without encoding. If no security protection is provided in the lower level protocols, it is convinient for attackers to intercept PVM messages[4,5];

2) Neither is TID included in PVM messages encoded. It is an easy job for attackers to pretend to be the legal PVM task sending messages to other PVM tasks;

3) Attackers can attack message source by forging one or a bat of TIDs, according to the way TID is constructed;

4) Since no integration inspection exits in PVM message, it may be sent to destinate PVM task after being intercepted or interfered by attackers;

5) Lack of time stamp , messages may be intercepted and replayed at some specific time.

```
sendmessage(msg) {
 check wheter security parameters are set;
if ( security parameters ) {
read security parameters;
handly msg with the security parameters set by
users; }
call sendmessage() to send msg;
}
```

Fig. 1 sendmessage() algorithm

2 Realization of A Secure PVM

As the above reasons, the present PVM does not have a good security protection: attackers can intercept or interfere messages delivered among PVM tasks, and send messages to other PVM tasks in the name of legal PVM task. To enhance PVM security, this paper provides a security mechanism specific to PVM itself.

2.1 Introduction of the Fundimental Design

The main purposes of the security scheme are: 1) providing a security mechanism to prevent PVM from being intercepted, interfered or replayed; 2) providing a configuration mechanism, and enabling users to halt or start part or whole of PVM security functions, to make private key consultance; 3) independent of the lower level protocols of networks.

By analyzing the inner working mechanism of PVM, internal functions responsible for message sending, like sendmessage(), pvm_send(), pvm_mcast(), pvm_psend(), and message receiving functions, netentry(), pvm_recv(), pvm_nrecv(), pvm_trecv(), are adapted (See Fig.2 and Fig.3). Each message will be handled by the internal message sending functions before sending out, according to current security configuration. And vice versa to message receiving procedure, so that those messages that fail to pass the inspection will be disgarded.

To ensure users' abilities to configure PVM security mechanism, security PVM system provides two new PVM callings: pvm_setsecurityopt() and pvm_getsecurityopt(). The first process of PVM task can call pvm_setsecurityopt() to set capture property before other processes are forked. Also, pvm_getsecurityopt() may be called to get security property of the current task by any process during the run time.

2.2 Message Source Authentication Algorithm

The functions of message source authentication lie in: 1)confirmation of the authenticity of message source; 2) ensuring no changes are made during message delivery. We choose message abstract algorithm MD5 to be the algorithm for message source authentication. The authentication process operates as follow: management station and proxy server share a common authentication secret key. Before sending data, the sender make encoded PVM messages, time stamp and authentication secret key be the input, get an abstract value using MD5, and then send the message to the receiver with the abstract value in the corresponding field of the head of data package. When message arrives, the receiver does exact the same process as the sender, get an abstract value, and compares it with that included in the head of message package. If the two values are not equal, it means: 1) the message source is not authentic; 2) or the message has been altered.

```
receivemessage()
{ call receivemessage() to get msg;
check wheter security parameters are set;
if ( security parameters ) {
read security parameters;
        handly msg with the security
        parameters set by users;
}
if ( fail to pass security inspection )
disgard msg
}
        return msg to user;
}
```

Fig 2 receivemessage() algorithm

2.3 Algorithm of Preventing Message Replay

Besides confirmation for authenticity of message source, promptitude (without delay or replay) and alignment relationship of the messages delivered are also desired. The algorithm preventing message replay is rather complex. Its main idea is: a time stamp is contained in the head of data package (illegal modification of the value will cause the message to be disgarded, since the value of time stamp is one of the parameter of MD5 message abstract algorithm), whose value is maintained by the loose synchronous clock of the individual computers in the PVM. With it, receivers can judge the correct order of messages and their promptitude.

2.4 Divulge Preventing Algorithm

To prevent messages being divulged, encoding is required. Only receivers granted are able to recover the initial information. Here, we choose DES-CBC to be our encoding algorithm. As to the deployment of secret key, either the same secret key shared by all computers on the PVM, or by processes consist a PVM task is legal. The senders encode PVM messages in the data packages before delivering, while receivers decode them.

3 Conclusion

With the widespread of PVM application, PVM security issue becomes more and more prominent. To improve PVM security, it is useful to take the advantage of the security property of the lower level protocols of the networks, such as Ipv6. However, the method has its limits,since PVM is a heterogenous platform supporting multiple networks protocols. And as different users' requirements of PVM security vary greatly, dynamic configuration of security mechanism is necessary.

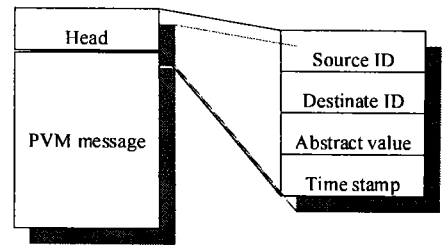

Fig.3 Format of data package

In this paper, a PVM security scheme integrades multiple security algorithms, which is able to prevent various attacks to PVM, independent of the lower level protocols of networks, and dynamic to the use of clients. It is a feasible choice to those who want to enhance PVM security mechanism.

References

1 林水生, 黄顺吉. 一种面向 MIMD 并行机实现的 FFT 并行算法. 电子科技大学学报, 1997, 26(6): 621~626

2 Geist Al, Beguelin Adam. PVM: parallel virtual machine. California: The MIT Press, 1994

3 Blumenthal U, Wijnen B. User-based security model (USM) for SNMPv3. RFC2264, 1998

4 Zhou Mingtian, Wang Wenyong. TCP/IP network theory and technology. Bei Jing: Tsinghua University Press, 1993

5 Atkinson R. Security architecture for the internet protocol. RFC 1825, 1995

一种 PVM 安全机制的设计与实现*

薛 兰** 宋 杰 陈锡明

(电子科技大学计算机学院 成都 610054)

【摘要】 PVM 的安全机制是决定其大规模部署可行性的一个重要因素，文中提出了一种增强 PVM 安全性的方案。它解决了在构成集群的各个计算机之间传递消息所面临的被窃听、篡改、伪造和重放的安全性问题，独立于具体的下层网络协议，消耗系统资源少，并且具有可动态配置的优点。

关 键 词 并行虚拟计算机; 集群计算; 并行计算; 网络安全; 消息; 窃听; 篡改; 伪造; 重放

中图分类号 TP316.81

1999年12月6日收稿

* 信息产业部发展基金资助项目

** 女 23岁 硕士生