Recovery Model for Free-Roaming Mobile Agent against Multiple Attacks

S.Venkatesan
Department of CSE
Anna University
Chennai-600025, India
venkalt_s@yahoo.co.in

C.Chellappan
Professor, Department of CSE
Anna University,
Chennai-600025, India
drcc@annauniv.edu

ABSTRACT
In the free-roaming mobile agent system, the recovery model is complicated then the itinerary mobile agent systems and the distributed Database environment. This paper proposes the recovery model for free-roaming mobile agent and its collected offers by the k-response methods to improve the tolerance of the agent failure or occupied by the malicious host. In the k-response model it is able to identify whether the agent is migrated from one to another host or not. If it is not migrated than the genuine host resent the agent once again to the host next to the malicious host and mentioned in the list that the host which is not migrated the agent from them.

Categories and Subject Descriptors
C.2.0 **[General]** : Security and protection I.2.11 **[Distributed Artificial Intelligence]**: Intelligent Agent D.2.0 **[General]**: Protection Mechanism

General Terms
Security, Algorithms,Theory

Keywords
Free-roaming mobile agent, colluded attacks, agent recovery, k-response, offer recovery.

1.INTRODUCTION
Mobile agent environment is the emerging technical model to reduce the network traffic by sending them to the remote host to collect the information for the owner instead of sending and receiving the information between the local host and the remote host. Mobile agent is the software program, which is dispatched to the various remote hosts to collect the information on behalf of its owner (local host). The owner gives traveling plan of the agent. The next type of the agent is the free-roaming mobile agent; it roams across the network to collect the information to the owner without any traveling plan given by the owner. The current host has to decide the next host for the agent to travel.

MOBIMEDIA 2007, August 27-29, Nafpaktos, Greece

DOI 10.4108/ICST.MOBIMEDIA2007.1835

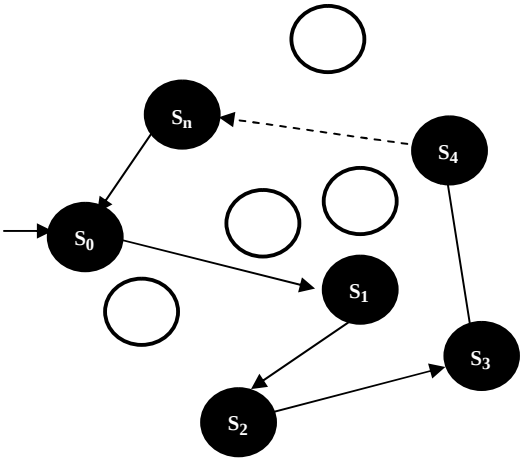

Fig.1. Agent Migration

During the migration of free-roaming mobile agent, the agent may get crashed or occupied and destroyed by the malicious host. This will cause the serious problem to the owner, because the owner of the agent waiting for the information from the agent to continue its next process. This situation requires recovery process for free roaming mobile agent.

This paper proposed the recovery model for the free roaming mobile agent. The discussion of this paper is the related work and notations and then the issues in the free roaming mobile agent and the various attacks and recovery model and cost evaluation and then conclusion

2.RELATED WORKS
Related works given in the below are the recovery model for the ordinary mobile agent environment. Silva and Popescu[1] describes an approach that relies on combining agent replication and transaction-controlled mobility to provide reliability for distributed agent applications. Their approach does not consider the state of the inter-agent communication channels upon recovery and thus does not cater for duplicate and out-of-door messages. Maintaining this property ensures that an agent recovered from a failed state does not re-execute an itinerary step steps previously performed by the multiple agent replicas[2]. This problem usually arises from imperfect detection of agent failure.

Eiter et. al[3] agent framework is designed to support situated agents, whose action results must be sensed, and for which failure must be explicitly detected.

Fedoruk et. al[4] describes a method for recovering from execution problems by backtracking to a diagnosed point of failure, based on execution monitoring, from which the agent continues towards its original plan. The backtracking is enabled by building a library or reverse plans corresponding to action sequences.

Morley et. al [5] describe a framework that incorporated server-level exception-handling and the use of process pairs in a mobile agent context. However, in their domain they do not address issues in updating the shadow with the primary's state after the initial replication.

Pears et. al[6] Prospose and approach to agent replication, which has similarities with the primary/shadow model. However, in their discussions of state replication and "switch over" they do not take into account the situated recovery issues addressed here.

Unruh et. al[7] describes the shadow model also gets the update where the primary model gets it. Then the rollback recovery will be done from the last update.

3.NOTATIONS

The various notations used in this paper are

Table.1 Notations

Notation	Meaning
Tt	Total time of the agent to roam
$Mt_{i,i+1}$	Migration time from host i to i+1
Vt	Verification time[9] for the offer in the host for reliability of the offers
Ct	Time for computing the offer (includes the offer generation and cryptography[9] applied to the offers)
Rt	Response time
Ht	Host time for verification and computation
n	Number of nodes agent going to visit
I_A	Agent identity
O_i	[9] Protected offer of the host i
$R_{i,I-1}$	Response from the host i to i-1

4. ISSUES IN FREE-ROAMING MOBILE AGENT

The owner of the agent sends it to collect information from the various remote host and waiting for the result of the agent. If the agent fails or occupied and destroyed in the middle of the path, the owner doesn't know about the failure. It is waiting for the return of the agent. This will waste the time of the owner and also waste the time of the remote host to compute its offer(partial result[9]). The offers collected in the previous host also lost.

Next the owner of the agent identifies that the agent crashed by exceeding of the given time to the agent to roam in the network. Once again it generates its agent and forward it to roam the network to collect the same offers. Then the remote hosts once again computes its offer and provide cryptographic mechanism[8] for the offers.

Assume that once again the agent may crashed or destroyed. Then this will be very sensitive issue of the owner. It is not able to identify the malicious host also. We can also say that the owner is not able to get the offer from the remote host by continues occurrence of crash in the route. Again the total time is wasted. This paper provides the solution for these issues in the Free-roaming mobile agent environment.

5. AGENT RECOVERY

This paper proposes the k-response method to do the recovery in the free- roaming mobile agent environment. That is the failure of the agent is identified by the response of the preceding hosts. We are considering here the various scenarios.

Scenario 1:

The agents have to migrate from one host to another host to collect the information. At the time of migration, the agent can be failed or crashed due to some interference. The offers collected in the previous host will lost and there is no further operation. This is depicted in the fig.2

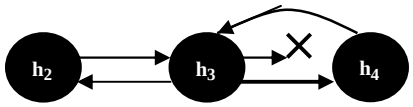

Fig.2: Agent failure during migration and recovery in 1-response method

The transition in the fig.2 is

$h_2 \rightarrow h_3 : O_0, O_1, O_2, I_A$

$h_3 \rightarrow h_2 : R_{3,2}, I_A$

$h_3 \rightarrow h_4 : O_0, O_1, O_2, I_A$ (but fails)

$h_3 \rightarrow h_4 : O_0, O_1, O_2, I_A$ (resending after failure identified)

$h_4 \rightarrow h_3 : R_{4,3}, I_A$

Here host h_3 forwards agent to h_4. Agent during migration gets failed or crashed. As per our k-response method, we use the 1-response method for this problem. The host h_4 have to respond the host h_3 with the next host identity and the agent identity after receiving the agent. If the agent is not received by the host h_4 , then there is no response to h_3. h_3 waits for a given period for first response, if it is not received the response means it identifies that the agent is failed during the migration. Then the host h_3 once again forwards the agent to the host h_4 to perform its operation. Each host has to maintain the copy of the agent and its offers until it receives the k-response from the preceding host. The host h_3 keep on watching for the 1-response to conform the agent is forwarded to the next host or not.

Here another issue can rise that is the host h_4 received the agent but it will not send response to the host h_3. This will make the host h_3 to forward the agent again and again. For this reason, the host have to forward the agent only two times to the host h_4 after that it

will change the route and forward the agent to the another host h_5. If h_4 destroys agent means it is identified then the report will be send to the owner of the agent as h_4 is malicious host. If the path makes the agent failure then the host h_4 will be visited later to collect the information from it.

Scenario 2:

In this scenario we consider the single host attack with two issues: 1.Host receives the agent but not send response to the sender. 2. Host receives the agent and sends the response to the sender but not forward the agent to the next host.

The first issue is discussed in the previous scenario. Here we will discuss the second issue as in fig.3. Host h_3 forwards agent to host h_4. h_4 receives the agent and send the response to the host h_3 that it receives the agent. After that it occupies the agent and keeps itself or destroys it. Gray arrow in the all the diagrams represents no transition.

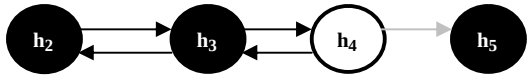

Fig.3: Agent destroyed by the single malicious host

The transition in the fig.3 is
$h_2 \rightarrow h_3 : O_0 , O_1 , O_2 , I_A$
$h_3 \rightarrow h_2 : R_{3,2} , I_A$
$h_3 \rightarrow h_4 : O_0 , O_1 , O_2 , O_3 , I_A$
$h_4 \rightarrow h_3 : R_{4,3} , I_A$
$h_4 \rightarrow h_5 :$ No transition or migration

If we use the 1-response method means we are not able to identify the attack. We have to use the 2-response method in fig. 4 to avoid this problem. That is the host h_3 forwards the agent to host h_4. Then the host h_4 have to respond the host h_3 with the host h_5 identity and the agent identity. After that the host h_3 will wait for the response from the host h_5. If the host h_4 receives the agent and keep itself means the host h_5 will not send the response to the host h_3. From this the host h_3 identifies the host h_4 is malicious and then it change the path and also mention the remarks of host h_4 in the host list[1]. Now the host h_3 sends the agent to host h_5.Thick arrow represents the retransmission.

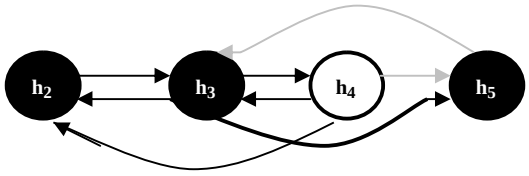

Fig.4: Agent destroyed by the malicious host and recovery in 2-response method

The transition in the fig.4 is
$h_2 \rightarrow h_3 : O_0 , O_1 , O_2 , I_A$
$h_3 \rightarrow h_2 : R_3 , I_A , h_4$
$h_3 \rightarrow h_4 : O_0 , O_1 , O_2 , O_3 , I_A$
$h_4 \rightarrow h_3 : R_4 , I_A , h_5$
$h_4 \rightarrow h_2 : R_4 , I_A , h_5$
$h_4 \rightarrow h_5 :$ No migration
$h_5 \rightarrow h_3 :$ No response
$h_3 \rightarrow h_5 : O_0 , O_1 , O_2 , O_3 , I_A$

Scenario 3:

In the free-roaming mobile agent environment, we have to mainly concentrate on the colluded attacks[9]. Here we discuss about the two-colluded attacks[9]. We find the two-possible way of attacks in two colluded attacks one is the two hosts are in adjacent place to destroy the agent and next is the hosts are in various place(not adjacent) to destroy the agent. The second type of colluded attacks will be avoided by the method in the scenario 2. We consider the first method in this scenario which is represented in fig.5

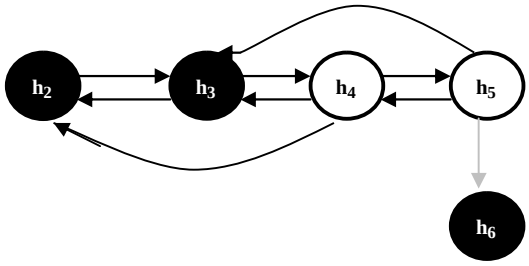

Fig.5: Agent destroyed by the colluded malicious host

The transition in the fig.5 is
$h_2 \rightarrow h_3 : O_0 , O_1 , O_2 , I_A$
$h_3 \rightarrow h_2 : R_{3,2} , I_A , h_4$
$h_3 \rightarrow h_4 : O_0 , O_1 , O_2 , O_3 , I_A$
$h_4 \rightarrow h_3 : R_{4,3} , I_A , h_5$
$h_4 \rightarrow h_2 : R_{4,2} , I_A , h_5$
$h_4 \rightarrow h_5 : O_0 , O_1 , O_2 , O_3 , O_4 , I_A$
$h_5 \rightarrow h_4 : R_{5,4} , I_A , h_6$
$h_5 \rightarrow h_3 : R_{5,3} , I_A , h_6$
$h_5 \rightarrow h_6 :$ No further process

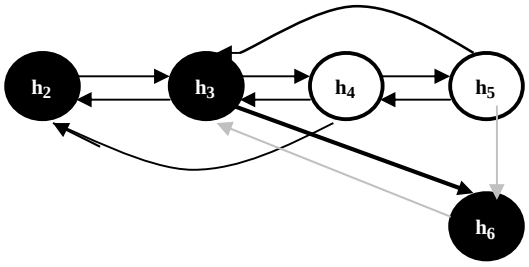

Fig.6: Agent recovery by 3-response method after the colluded malicious host attacks.

[1]List contains the visited host and then the next visiting host, which is provided in the initial stage. It makes the remarks for the malicious host in the list, which will be sent to the owner at last. Nodes before h_2 and h_5 and its migration and response are not represented here.

Host h_3 forwards the agent to the host h_4 and then h_4 sends the response to h_3. Then it forwards the agent to host h_5 then h_5 sends the response to h_3. Then the host h3 give the control to the host h_4.

The transition in the fig.6 is

$h_2 \rightarrow h_3 : O_0 , O_1 , O_2 , I_A$
$h_3 \rightarrow h_2 : R_{3,2} , I_A , h_4$
$h_3 \rightarrow h_4 : O_0 , O_1 , O_2 , O_3 , I_A$
$h_4 \rightarrow h_3 : R_{4,3} , I_A , h_5$
$h_4 \rightarrow h_2 : R_{4,2} , I_A , h_5$
$h_4 \rightarrow h_5 : O_0 , O_1 , O_2 , O_3 , O_4 , I_A$
$h_5 \rightarrow h_4 : R_{5,4} , I_A , h_6$
$h_5 \rightarrow h_3 : R_{5,3} , I_A , h_6$
$h_5 \rightarrow h_2 : R_{5,2} , I_A , h_6$
$h_5 \rightarrow h_6 :$ No further process
$h_6 \rightarrow h_3 :$ No Response
$h_3 \rightarrow h_6 : O_0 , O_1 , O_2 , O_3 , I_A$

Now the host h_4 is the colluded malicious host with host h_5, which will just drop its functionalities and no further operation is there. To avoid this situation,3-response method as in fig.6.The host h_3 forwards the agent and get the response from the host h_4 and wait for the response from the host h_5 and also waiting for the response from the host h_6. If the agent is attacked by the malicious host h_4 and h_5 then the response from the host h_6 will not received by h_3. Then h_3 forwards the agent once again to h_6 and mention in the list that the host h_4 and h_5 are malicious.

Scenario 4:

Next to the two-colluded attacks is the multiple colluded attacks[9] that is more than two malicious host combined together and destroys the agent. The colluded hosts may be in the adjacent place or in the various places. If it is in the various places (i.e., each host in separate place and the next is two host in adjacent place another one in some other place) means the scenario 2 and 3 will overcome this issue.

If all the nodes are in the adjacent place means we are not able to use the k-response methods because the waiting cost of the remote host will increase and the network gets more traffic. For this we have to verify the environment, whether it is genuine or not. If it is not genuine then the travel has to be avoided. This is being verified in future.

6. COST EVALUATION

The time evaluation to collect the offers by the free roaming agent in the environment is depend upon the migration time, verification time for the reliability[9] and the computational time of the host to generates its offer and to provide crypto graphical model. The total time taken to collect the offer from the network with n host is

$$Tt = \sum_{i=1}^{n} Mt_{i,i+1} + Vt_i + Ct_i \longrightarrow (1)$$

time taken by the each host to computes its offer is *Ht.*

$$Ht = Vt + Ct$$

Time calculated in the above is less because there is no response time but there is no surety that the owner will receive the offer or not. If there is any attack then the owner never get the offers. From the recovery model given in this paper we will sure that the offer will reach the owner. The total time taken by the agent in the each response model is the mentioned in below equations. The total time taken by the agent to collect the information after the 1-response recovery model is

$$Tt = \sum_{i=1}^{n} (Mt_{i,i+1} + Vt_i + Ct_i) + Rt_4 \longrightarrow (2)$$

Here the value of $i=1,2,3,4,5,6,\ldots n$

The total time taken by the agent to collect the information after the 2-response recovery model is

$$Tt = \sum_{i=1}^{n} (Mt_{i,i+1} + Vt_i + Ct_i) + Rt_5 + (Mt_{4,5} + Vt_4 + Ct_4) \longrightarrow (3)$$

Here the value of $i=1,2,3,5,6,\ldots n$

The total time taken by the agent to collect the information after the 3-response recovery model is

$$Tt = \sum_{i=1}^{n} (Mt_{i,i+1} + Vt_i + Ct_i) + Rt_6 + (Mt_{4,5} + Vt_4 + Ct_4) + (Mt_{5,6} + Vt_5 + Ct_5) \longrightarrow (4)$$

Here the value of $i=1,2,3,6,7,\ldots n$

Agent will not wait for the response time between the hosts. It carried outs its process until there is no attack. If there is attack, there is a need for waiting time Rt otherwise the time taken by the agent to roam in the network is same as (1). The equations in the above are for the one-time attack. If it exceeds the Rt also exceeds.

7. CONCLUSION

This recovery method in the free-roaming mobile agent will recover the lost offers from the various attacks. Also it identifies the malicious host and inform to the owner to take the necessary action against the malicious host. This is helpful for the safety of the offers collected from the various hosts in the distributed environment. The time of the owner or creator and the time of each remote host are fully saved by this recovery model instead of time taken to once again forwarding the agent and computing its offer and provide the security model in the offers. This model makes the time useful.

8.ACKNOWLEDGMENT

This Work is supported by the NTRO, Government of India. NTRO provides the fund for collaborative project "smart and secure environment" and this paper is modeled for this project. Authors would like to thanks the project coordinators and the NTRO members.

9. REFERENCES

[1] F.Silva and R.Popsecu-Zeletin, "mobile agent-based transaction in open environments." *IEICE/IEEE JOINT Special Issue Autonomous De-centralized Systems*, vol.E83-B,no.5,pp. 973-987,may 2000.

[2] M.Breugst, I.Busses, S.Covaci, and T.magdanz, "*Grasshopper-A mobile agent platform for IN based service environments", in Proc. IEEE Intelligent Networks Workshop*, Bordeaux, France,May 1998,pp. 279-290.

[3] T.Eiter, E.Erdem, and W.Faber. Plan reversals for recovery in execution monitoring. In *Non-monitoring Reasoning,2004.*

[4] A.Fedoruk and R.Deters. Improving fault-tolerance by replicating agents. In AAMAS'02,pages 737-744. ACM Press,2002.

[5] D.Morley and K.Myers. The SPARK agent framework. In AAMAS'04, NY, NY, 2004.

[6] S.Pears, J.Xu, and C.Boldyreff. Mobile agent fault tolerance for information retrieval application: An exception handling approach. In *The Sixth international Symposium on Autonomous Decentralized Systems, 2003.*

[7] A.Unrh, H.Harjadi, J.Bailey. *Semantic-Compensation-Based Recovery in Multi-Agent Systems*. In IEEE.2005

[8] V. Roth. "On the robustness of some cryptographic protocols for mobile agent protection." In *Proceedings of the 5th International Conference on Mobile Agents (MA 2001)*, volume 2240 of *Lecture Notes in Computer Science*, pages 1–14. Springer-Verlag, 2001.

[9] D.Xu, L.Harn, M.Narasimhan, J.Luo. "An Improved Free-Roaming Mobile Agent Security Protocol against Colluded Truncation Attacks." *In Proceedings of the 30th Annual international Computer Software and Applications Conference(COMPSAC,06),* Volume 2, Page(s):309 – 314 ,Chigago,Sept. 2006,IEEE Computer Society Press.