IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 2, February 2016

A Novel Approach to Compute Reputation Value for Trust based Hybrid Security Architecture for Mobile Agents

Dr. Heman Pathak

Associate Professor, Dept. of Computer Science, GKV, Haridwar, Uttarakhand, India

Abstract: Reputation and trust management systems have been useful in situations that involve interaction between mutually distrusting parties to function correctly and to fulfil their purposes. Trust Management System based on reputation gained popularity in recent time for estimating the trustworthiness and predicting the future behaviour of nodes and other network entities in a large-scale distributed system. Trust and reputation have gained importance in diverse fields such as economics, evolutionary biology, distributed artificial intelligence, grid computing, and agent technology, among others. Many researchers have proposed different approaches to compute reputation/trust value to evaluate the trust worthiness of the entities involved. Approaches mainly differ based on the system model they have used, entity for which trust worthiness is computed and area of application. This paper explores the trust model based on reputation to provide security to both MA and executing host. In order to establish safe and secure communication between agent and hosts, each must be trusted that it would not harm other when it is given the access in a system. Paper presents a new way to compute reputation value of both host and mobile agent based either on past experience or experiences of other trusted and known entities and third party. Reputation of host is evaluated by the previous experience of the MAs being executed on the host and also by using intrusion detection mechanisms.

Keywords: Mobile Agents (MA), Mobile Agent Systems (MAS), security, Intrusion Detector System.

I. INTRODUCTION

Mobile Agent (MA) is a software program that lives in computer networks, performing its computations and moving from host to host as necessary to fulfil user goals [11]. MA technology has gained popularity in the world of computing to enable easiest way of service provisioning, information sharing and service recovery, but securing MAs, as they move from one machine to the other, has been the challenge that hinders full adoption of this technology by organizations [5]. Autonomous behaviour of MA and the malicious environment of the internet give rise to various important security issues related with both MA and its host [1]. There are various security attacks identified by researchers on Mobile Agent Systems (MAS). The major security threats are either threats against the hosts, or threats against the MA [2].

II. ELEMEMTARY SYSTEM MODEL

In my previous papers [12][13], I have already presented Hybrid Security Architecture (HAS) for MA. My work is inspired by the various existing security techniques including digital signature, encryption, intrusion detections, signed agreement, trust and others [3] [4].

Since all are well studied and experimented techniques, paper does not discuss all in details. Proposed approach divides the open network like internet into regions and then assigns the responsibility to one of the centralized component within the region to implement security features to protect malicious host and MA from each other. Instead of doing a logical partitioning of the network into regions and then arranging these regions into hierarchy, I use the existing technology to serve our purpose.

Internet is network of networks. Networks are connected with each other via router [11]. Proposed approach treats each network as a region and router as the centralized component in each region. Router in proposed architecture is not passive but plays an active role. A MA wishes to visit a host within a network, first arrive at the router of the network and then only pass to the designated host. Host in a network offer services and provide an executing environment to the MA to be executed. A host may be malicious and may tamper the executing MA. Similarly a malicious MA can harm the host in many ways so both need to be protected from each other. Routers are fault-free and trustworthy. In each network there is a shared local storage space (LSS), assumed to be fault free and trust worthy.

III.SYSTEM COMPONENTS

Internet is network of networks and in each network Hardware components of the system involve in identifying the malicious MA/Host and providing security to them are Router, Agent Server/Host and Local Shared Storage Space. Figure-1 shows the elementary model of HSA and various its components.

A. Router

Each MA enters in to a local network or migrates from the network via router. Router is a centralized component within each network. It is responsible to detect a malicious host within the network. It also checks the status of

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 2, February 2016

incoming MAs as malicious or not and block the malicious MA. Software components other than MAS installed at router and their descriptions are discussed in the following section.

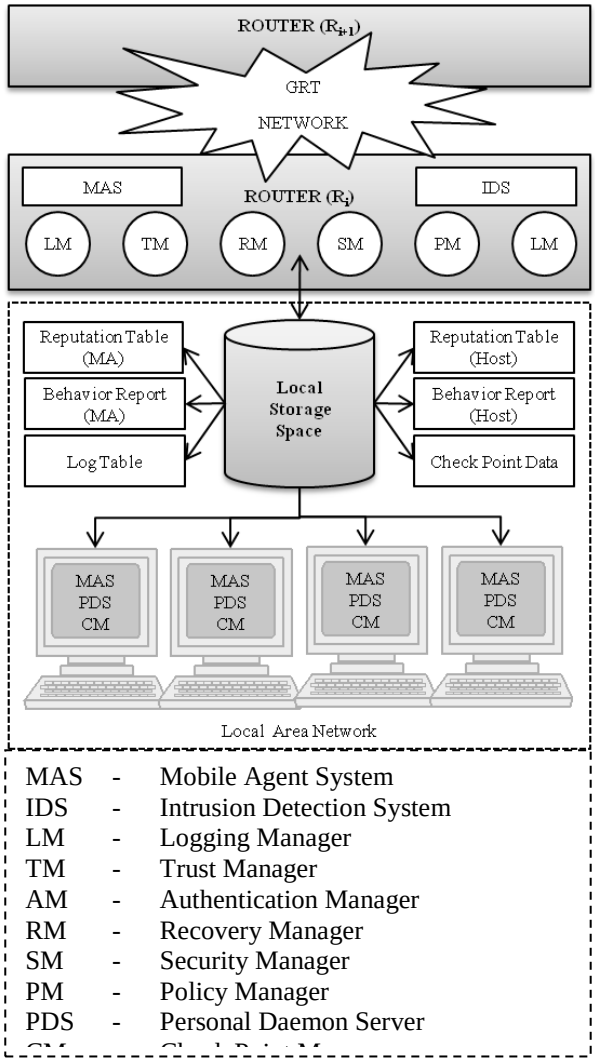

Figure 1: Trust based Hybrid Security Architecture (HAS)

1) Intrusion Detector System (IDS)
I propose a centralized approach to detect the malicious host within the network. An IDS installed at the router randomly creates intruder detectors (ID) and execute it on various hosts and record their behavior. Behavior report is then analyzed and RV is updated for host.

2) Logging Manager (LM)
This s/w routine is responsible for Log an arrival and departure entry in log table for each MA received and migrated from the network respectively. These log entries are used for tracing the MA path and for recovery from a fault state.

3) Trust Manager (TM)
This s/w routine installed on router is responsible for computing reputation value (RV) for all incoming and outgoing MA via router. It also maintains the reputation value of the hosts, part of the network [5][6].

4) Authentication Manager
Once a MA is found trustworthy, this s/w component checks access rights of MA and authenticate MA based on digital signature. It also verifies its code and data by using public and private key mechanism.

5) Recovery Manager
This s/w routine installed at router is responsible to initiate recovery procedure in case a MA or host is found malicious. Recovery procedure is quite complex and lots of policy and legal issues are involved with it, so I am not discussing them in this paper but left for future work.

B. Host
Host is a computer in the network which offers services to the MA. It provides executing platform to the MA. A MA is passed to be executed on a Host only if both MA and host are found trustworthy. To ensure the trustworthiness of both, some of the security components are installed at the host. Following section discuss these components.

1) Personal Domain Server (PDS)
PDS is a proxy server, installed at each Host. It maintains a thread to watch the behavior of the host. When a MA is arrived at a host for execution, it starts threads to record the behavior of MA and executing platform. Executing After the execution of MA, it prepares and store reports for at the local shared storage space.

2) Checkpoint Manager (CM)
This is responsible to save the MA and its execution state to LSS periodically and after every successful transaction. This checkpoint data can be used to recover the host in case it has been attacked by the malicious host. These data can also be used to recover the MA in case malicious host attacked it.

C. Local Shared Storage Space (LSSS)
Each network is assumed to maintain a fault free storage space. This space is accessible by all hosts and components installed at router. It is used to store Log Table, behavior report of MA and Hosts, reputation table for MA and hosts.

D. Global Reputation Table
Hosts are static component of the system and its behavior can be watched locally while MA is a moving entity and its behavior is watched by various components distributed across the network. To compute the reputation value (RV) for the MA, observations of each entity interacted with MA must be compiled. For this purpose I propose to maintain a Global Reputation Table (GRT) for MAs. GRT is maintained by some server in the global network.
This table is accessible to all the routers and assumed to fault free and trust worthy. Since accessing and updating this table is time consuming and will increase lots of network traffic, this table only maintains the list of MAs and their RVs that have been found suspicious or malicious by some watching entities. This table is concerned only when information gathered locally or from source router of MA is insufficient to make decision about the RV of the MA.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 2, February 2016

IV. REPUTATION VALUE COMPUTATION

Various researchers have proposed various ways to compute RV based on their model and application [7] [8] [9] [10]. Based on the experience gained by the prior work I propose a new way to compute RV for MA and hosts. RV for MA is collected by various components of the system and then appropriate weights are assigned to each RVs. Mathematical computation among RVs and weights are used to compute new RV. Table -1 shows the RV assigned to an agent or host and their interpretation. RV is assumed to be in the range [-1, 1].

TABLE1. REPUTATION VALUES ASSIGNED AND THEIR INTERPRETATION

RV	Meaning
-.7 to -1	Malicious
.3 to -.6	Suspicious
-.2 to .2	Average
.3 to .6	Trusted
.7 to 1	Highly Trusted

Figure-2 shows the various components of the system and their interaction with TM to compute the RV. Movement of MA has also been displayed.

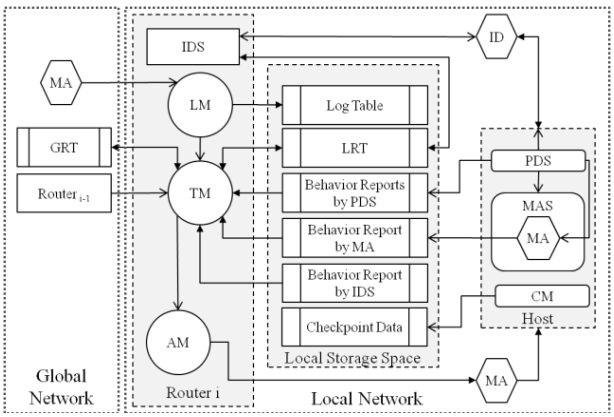

Figure 2: Interaction of TM with other components to compute RV

A. Reputation Value Computation for Host

TM is responsible to analyze the behavior report of the host submitted by the IDS, executed MA and PDS. Based on these reports TM computes new RV for the host and update the LRT for host. An initial RV for a host is 0. Host RV is computed locally and stored only at LRT as $RV_h(OLD)$. Different symbols used for different RVs shown in Table-2.

TABLE2. SYMBOLS USED TO REPRESENT DIFFERENT RVS FOR HOST

$RV_h(NEW)$	:Newly Computed RV of the Host.
$RV_h(OLD)$:	RV as stored in the LRT of host.
$RV_h(PDS)$:	RV based on behavior report by PDS.
$RV_h(MA)$:	RV based on behavior report by MA.
$RV_h(IDS)$:	RV based on IDS observations.

In order to compute the new RV for a host based on various RVs, some weights have been assigned to each RVs. Symbols used different weight is shown in Table-3.

TABLE3. SYMBOLS USED FOR WEIGHTS ASSIGNMENTS AT HOST

W_{ho}	: Weight assigned to $RV_h(OLD)$.
W_{hp}	: Weight assigned to $RV_h(PDS)$.
W_{hm}	: Weight assigned to $RV_h(MA)$.
W_{hi}	: Weight assigned to $RV_h(IDS)$.

An IDS installed at router observes the behavior of host periodically by the method discussed earlier. IDS then compute $RV_h(IDS)$ for host based on observations. This value is then used to update the RV of host in LRT as -

$$RV_h(NEW)=W_{oh}*RV_h(OLD)+W_{ih}*RV_h(IDS)$$

The choice for W_{ih} is critical. It will depend upon how efficiently and frequently IDS has been implemented, what behavior has been observed and which analyzing techniques have been used. W_{oh} is then selected accordingly such that $W_{ih} > W_{oh}$. $RV_h(NEW)$ is then scaled to bring in the range (-1 to 1) and LRT for host is updated.

Trust Manager installed at router computes the RV for the host based on behavior reports submitted by PDS and executing MA. TM analyzes these reports and compute two RV for host as $RV_h(PDS)$ and $RV_h(MA)$. Now different weighs are assigned to different RVs. Maximum weights are assigned to the newly computed values. RV of the MA is used as weight for $RV_h(MA)$ as –

$$W_{hm} = RV_m .$$

Values for W_{oh} and W_{ph} then selected such that ($W_{ph} > W_{mh} > W_{oh}$). TM then compute $RV_h(NEW)$ as-

$$RV_h(NEW)=W_{oh}*RV_h(OLD)+W_{ph}*RV_h(PDS)+W_{oh}*RV_h(MA)$$

$RV_h(NEW)$ is scaled to bring in the range (-1 to 1).

B. Reputation Value Computation for MA

The RV for a newly created MA is same as the RV of its creator host as $\mathbf{RV_m = RV_h}$. Different symbols used for RVs and weights for MA is shown in Table-4.

TABLE4. SYMBOLS USED FOR RV AND WEIGHTS FOR MA

$RV_m(NEW)$: Newly Computed RV of the MA.	
$RV_m(OLD)$: RV as stored in the LRT of MA.	
$RV_m(PDS)$: RV based on behavior report by PDS.	
$RV_m(LVR)$: RV returned by the Last Visited Router.	
$RV_m(GRT)$: RV as stored in Global Reputation Table.	
W_{mo}	: Weight assigned to $RV_m(OLD)$.
W_{mp}	: Weight assigned to $RV_m(PDS)$.
W_{mr}	: Weight assigned to $RV_m(LVR)$.
W_{mg}	: Weight assigned to $RV_m(GRT)$.

DOI 10.17148/IJARCCE.2016.52146

1) RV computation for outgoing MA

For outgoing MA, TM is responsible for analyzing the behavior report of the MA, and based on this analysis update the RV of the MA in the LRT of MA. If TM found the behavior of MA malicious, it passes the MA to RM to initiate the recovery procedure. When a MA is ready to migrate, TM computes $\mathbf{RV_m}$(PDS) based on the report submitted by the PDS. RV of the host has been used as weight for RV_m(PDS) as $W_{mp} = RV_h$. Appropriate values for W_{mo} is selected such that ($W_{mp} > W_{mo}$). TM then compute RV_m(NEW) as-

$$RV_m(NEW)=W_{mo}*RV_m(OLD) + W_{mp}*RV_m(PDS)$$

RV_m(NEW) is then normalized to bring it in the range (-1 to 1) and LRT for host is updated. In case MA found suspicious or malicious, recovery procedure is initiated.

2) RV computation for incoming MA

For an incoming MA, TM collects the RV of the MA from the last visited router as $\mathbf{RV_m}$(LVR). MA may carry its RV with it but a malicious MA may alter its RV and can't be trusted. Since I have assumed that routers are trust worthy so RV collected from them are also trusted. Local Reputation Table (LRT) for MA is also consulted to get the RV of incoming MA if it has previously visited the network. If these data are not sufficient then GRT is consulted to check whether MA has been tagged as suspicious or malicious by any of the entities. TM then analyze these data and compute new RV for the MA and if found trusted pass it to the authentication Manager (AM). If MA is found suspicious or malicious, it is transferred to Recovery Manager. TM computes the RV_m(NEW) as average of RV_m(OLD) and RV_m(LVR). If no old RV_m is found or if data are not sufficient to make decision, GRT is consulted to check if MA suspicious or malicious, RV is then modified accordingly.

V. CONCLUSION

In the proposed architecture, only trusted MAs are transferred to the host and host gets protected from the attack of malicious MA. Also during the execution, behavior of MA is recorded and CM saves the MA and its execution state in the LSS periodically. In case MA attacks the host during execution, this attack can be detected and RM can use the checkpoint data to bring the host in consistent state. Since MA is allowed only to be executed on trusted host, it gets protected from the attack of the malicious host. Even during the execution if it has been attacked, RM can rollback all MA execution and recover it from checkpoint data.

I have proposed an architecture which logically secure the MA and Host both from malicious attack. Various components of the system work collectively to provide solution to the said problem. Since the proposed architecture has yet not been implemented or modeled, its practicality is still to be tested. Since most of the approaches used here are well known and has already been implemented successfully so it is quite reasonable to accept that, this architecture once implemented will solve the concern issues successfully. Its efficiency or comparative performance analysis is possible only after the implementation.

REFERENCES

[1] Ahmed Patel, Wei Qi and Christopher Wills, 2010. A Review and Future Research Directions of Secure and Trustworthy Mobile Agent-based E-marketplace Systems. Information Management and Computer Security, Vol. 18, issue No. 3, pp xx-xx, 2010.

[2] Ananta Charan Ojha, Sateesh Kumar Pradhan, Manas Ranjan Patra,2008. Demacom: A Framework for Developing Mobile Agent-Based E-Commerce. The Icfai University Journal of Information Technology, Vol. 4, No. 4, pp. 7-22, December 2008.

[3] De Capitani di Vimercati, S.Foresti, S.Jajodia et al. Integrating Trust Management and Access Control in Data Intensive Web Applications,ACM Transactions on the Web (TWEB) 6,2, 1-44 (2012).

[4] Habib S., Ries S., Muhlhauser, M., Towards a Trust Management System for Cloud Computing, In Proc. of IEEE 10th Int. Conf. on Trust, Security and Privacy in Computing and Communications (TrustCom'11). Changsha, China (2011)

[5] Jean Evens, Jiao Yu, and Hurson Ali R., 2007. Addressing Mobile Agent Security through Agent Collaboration. Journal of Information Processing Systems, Vol.3, No.2, December 2007 43 10.3745/JIPS.2008.3.2.043

[6] K. Hoffman, D. Zage, C. Nita-Rotaru, A survey of attack and defense techniques for reputation systems, ACM Computing Surveys 42 (1)(2009) 1–31

[7] M.T Nkosi, M.O Adigun, J.O Emuoyibofarhe, Agent-To-Agent Reputation-Based Trust Management, IADIS International Conference Applied Computing 2007.

[8] Nkosi M.T, Adigun M.O., Emuoyibofarhe J.O., 2007. Agent-To-Agent Reputation-Based Trust Management. IADIS International Conference Applied Computing 2007.

[9] Noor T. H., Sheng Q.Z., Credibility-Based Trust Management for Services in Cloud Environments, In Proc. of the 9th Int. Conf. on Service Oriented Computing (ICSOC'11). Paphos, Cyprus (2011).

[10] O. Onolaja, G. Theodoropoulos R. Bahsoon,, A Data-Driven Framework for Dynamic Trust Management, Procedia Computer Science Procedia Computer Science 4 (2011) 1751–1760.

[11] Patel, R.B. 2004. Design and implementation of a secure mobile agent platform for distributed computing', PhD Thesis, Department of Electronics and Computer Engineering, IIT Roorkee, India, Aug.

[12] Pathak H., A Novel Hybrid Security Architecture (HSA) to provide security to Mobile Agents and the Executing Host, Proceedings of the International Conference on Communication, Computing & Security Pages 499-502, Rourkela, 2011.

[13] Pathak H., A Novel Flexible and Reliable Hybrid approach to provide Security to Mobile Agents and the Executing Host, Proceedings of the International Conference on Electronics, Information and Communication Systems Engineering (ICEICE-2010), Jodhpur.