Secure Mobile Agent based Information Gathering in Wireless Network

Ashish Kumar Srivastava
Department of IT
India

Aditya Goel
Department of Electronics MANIT , Bhopal,
Engg. MANIT, Bhopal, India

Abstract- Nowadays, everything is moving towards the wireless environment to bring the smartness to the society. In this situation, it is necessary to bring the smart technologies in the wireless environment. By considering this in mind, we concentrated to incorporate the mobile agent in the wireless environment to gather information. The problem with the mobile agent (multi hop mobile agent) is the security issue in gathering information from number of remote hosts. To overcome this security issue, an 3-ID algorithm is available which will verify the integrity of the data as well as provide confidentiality to the data. But this algorithm requires more time complexity for verification of the previously collected all information integrity. To optimize the verification time complexity, this 3-ID algorithm [9][10] is modified to verify only N, N/2, N/3 or N/4 previous host information based on the requirements. The experimental results in the wireless environment proves that the verification time of the integrity will obviously less when compare to its original model.

***Keywords- Mobile Agent, Wireless network, Mobile Agent Security, Information Gathering*.**

1. Introduction

Wireless network has an significant impact on the today's network environment. It needs an time optimized technologies to incorporate into it as well as it needs an excellent security mechanism. This is the major motivation to incorporate the mobile agent concept in the wireless network

The mobile agent [4] is a software agent acting on behalf of its owner with the extra capabilities of mobility. Mobility refers to the migration of the agent from one node to another node to perform certain computations on behalf of its owner. The mobile agent that returns home after visiting more than one remote host in the single dispatch from the home is referred to as the multi-hop agent. The key reasons for incorporating the mobile agent concept in various applications are:

- Reduction of the network load: Mobile agents will reduce the data flow in the network by packing the conversation and dispatching it to a destination host for the agent to compute. The main advantage of the mobile agent is that the computation moved to the place where the data is available.
- Reduce network latency: Critical real-time systems, such as robots in manufacturing processes, need to respond to real-time changes in their environments. Controlling such systems through a factory network of a substantial size, involves significant latencies. For critical real-time systems, such latencies are not acceptable. Mobile agents offer a solution, because they can be dispatched from a central controller to act locally and execute the controller's directions directly.
- Dynamic Adaption: Mobile agents can sense their execution environment and react autonomously to changes.
- Robust and fault-tolerant: If a host is being shut down, all agents executing on that machine are warned and given time to dispatch and continue their operation on another host in the network.
- Client Customization: In distributed computing models like Remote Procedure Call (RPC) and distributed objects (RMI), the exposed functions are defined and established on the server and there is no opportunity for client customization. Clients are confined to the service provided by the server. In case the clients want to have a new service, the service must be installed on the server. But with the concept of mobile agent, the clients are virtually installing programs on to the server when the mobile agent migrates from one host to the others.

Despite its many practical benefits, users of the mobile agent technology suffer from various issues, mainly, security threats. To protect against these types of attacks, various solutions were developed, but the developed security models for a mobile agent environment do not give a guarantee to protect it from new types of attacks, and also no recovery model is reported for the multi-hop mobile agent.

The remaining section of this paper is organized as follows: Section 2 gives the brief description over the related works

and their impact in the intended environments. Section 3 describes about the existing 3-ID algorithm. Section 4 describes the proposed complexity mitigation model for integrity check and the experimental results of the extended model with its original model. Section 5 concludes the paper with the directions of future enhancements.

2. Related Works

Generally, a multi-hop mobile agent will visit more than one remote host in the single departure from the owner. The order of visiting the remote host may be static (travel path is given by the originator) or dynamic (travel path decided by the remote hosts). In both these cases, information protection is the major challenge against the attackers. During the journey of the agent, a single host (server) or a set of malicious hosts (servers) can collude together and modify, delete or insert malicious data (information or offers) in data set collected from the preceding hosts. For this, Yee [2] proposed the PRAC (Partial Result Authentication Codes) to protect the mobile agent information. Yee classifies his algorithm into three types:

i) Simple MAC-based PRACs

ii) MAC-based PRACs with one-way functions

iii) Publicly Verifiable PRACs

These three types of PRACs are the key associated. The key of the current host will be erased by the mobile agent prior to migrating to the next server or host. The agent has the list of encryption keys for each server to be visited. Even though the PRAC scheme ensures date integrity, agents must determine how many keys they need to carry before leaving the owner. Also, the agent has to carefully protect the keys and erase the used key once they complete their actions on each server. PRAC only provides weak forward integrity [1]. Also, this is impossible for multi-hop mobile agents in a real network environment [9][10].

Zhou et al [6][7] analyzed and developed a protocol to overcome the weakness of the Cheng et al model to defend two-colluder truncation attack. The Zhou et al [7] model is the same as the Cheng et al [5] scheme and uses a co-signing mechanism in which a host needs the preceding host's signature on its encapsulated offer before sending it to the next host. Even though it is able to protect the data from the two-colluded attack, it cannot defend against multiple-colluder (more than two) truncation attacks.

Xu et al [8] proposed a protocol to defend against the two-colluder truncation attack with the help of the one hop backward and two-hop forward chaining method. It also defends against the multiple colluder truncation attacks, fake stem attack and then the interleaving attack. It will defend the multiple colluder truncation attacks as long as any two of the colluders are not adjacent. This protocol can be extended to overcome the adjacent attacker limitation, but the protocol process will be too complex to implement. After this, 3-ID algorithm [9][10] is developed with complete security for the information in the mobile agent environment. The following gives the complete description of the algorithm.

3. 3-ID Algorithm

To protect the offers or data or information against colluded attacks, the proposed 3-Identity verification algorithm builds a chain relation and forwards it to the succeeding host just like the existing works but it has different computations. The computations are: (i) compute and encapsulate the offers with the identity: Every host will encapsulate its encrypted offers with its identity, the preceding host and the succeeding host identity and (ii) verify the integrity of the identity: Every host should verify the integrity of the identity of every encapsulated offer of the preceding host to identify the attacks. The reason for having the 3-Identity verification is to avoid the colluded, interleaving, fake stem and revisiting attacks. Block diagram for secure data transfer in wireless medium is given in fig1.The functions of every host in the multi-hop mobile agent environment are as follows:

(a) **Agent at the Creator (S_0):** The creator or owner creates and starts the agent with the dummy offer o_0 (to check from the return agent whether this item is modified or not) and random number r_0. Also it has to sign the dummy offer $Sig_{pr0}(o_0)$ for non- repudiation purposes and to encrypt those data for confidentiality Enc_{Pb0} ($Sig_{pr0}(o_0)$, r_0) with the help of the public key Pb_0 and forward it to the succeeding host S_1 . It also generates the temporary public and private key (tPb_0 , tPr_0) to sign the identity of the next host $Sig_{tpr0}(S_1)$. The function of the 3-ID algorithm in the agent's home is shown in below.

S_0 : Generate offer o_0
Compute $C_0 = Enc_{Pb0}(Sig_{pr0}(o_0), r_0)$
Generate tPr_0 , tPb_0
Decide next host S_1
$vh_0 = (H(Sig_{pr0}(S_0), Sig_{tpr0}(S_1), tPb_0 , r_0) \| (Sig_{pr0}(S_0), Sig_{tpr0}(S_1), tPb_0 , r_0))$
$O_0 = SigPr_0(C_0, vh_0)$
$S_0 \rightarrow S_1: O_0$

The hash function is applied to the encrypted identity, temporary public key and random number to identify the man in the middle attack. Every host in the network should have the public key of the other hosts. Finally, the agent originator S_0 signs the final encapsulated offer with its private key Pr_0 and sends its encapsulated offer (O_0) to the next selected host S_1 with the agent.

(b) **Agent at host or server S_1:** The agent from S_0 migrates to S_1 with the encapsulated offer O_0. After receiving the encapsulated offer, S_1 has to verify it for integrity and authenticity. The encapsulated offer is unveiled using the public key Pb_0 as follows.

S_1: Receive O_0
Recover C_0, vh_0 by Pb_0
Ver($Sig_{pr0}(S_0)$,$Sig_{tpr0}(S_1)$,tPb_0) - recover S_1 ,S_0

The identity of S_1 is recovered by tPb_0 and the S_0 is recovered by Pb_0; then, the identity sequence is verified from the initial offer O_0. In general, the first remote host has nothing to verify in the identity sequence but it has to check whether the identity of the current server is encapsulated in the protected offer or not. Next, the computation of the 3-ID algorithm at server S_1 is shown in below.

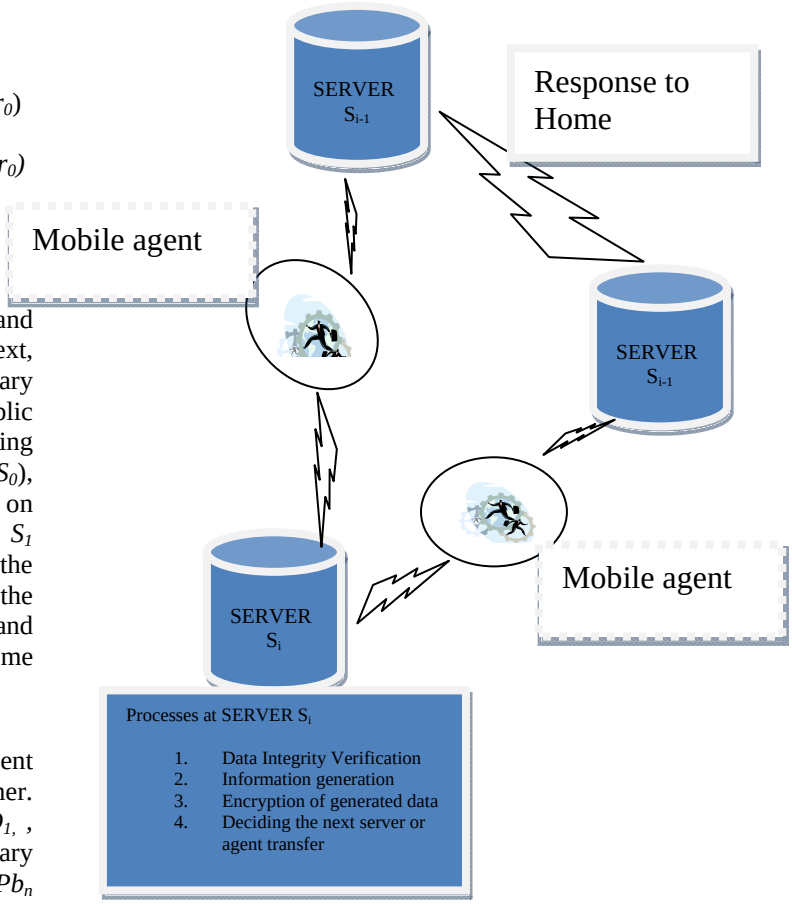
S_1: Generate offer o_1
Compute $C_1 = Enc_{Pb0}(Sig_{pr0}(o_1), r_1)$
Generate tPr_1 ,tPb_1
Decide next host S_2
$v\ h_1 = (H(Sig_{pr0}(S_0), Sig_{tpr0}(S_1),Sig_{tpr1}(S_2 ,tPb_0), tPb_1 ,r_0)$
||
$Sig_{pr0}(S_0), Sig_{tpr0}(S_1), Sig_{tpr1}(S_2 ,tPb_0), tPb_1 ,r_0)$
$O_1 = Sig_{tPr1}(C_1, vh_1)$
$S_1 \rightarrow S_2$: O_0, O_1, tPb_1

The S_1 will generate its offer o_1 and random number r_1 and then compute C_1 with the public key of originator Pb_0. Next, it has to encrypt the next server identity and the temporary public key of the previous server using the temporary public key ($Sig_{tpr1}(S_2 ,tPb_0)$, tPb_1 ,r_0) and append with the preceding server's last two identities $Sig_{pr0}(S_0)$, $Sig_{tpr0}(S_1)$ as $Sig_{pr0}(S_0)$, $Sig_{tpr0}(S_1)$,$Sig_{tpr1}(S_2 ,tPb_0)$, tPb_1 ,r_0 . Then it has to decide on the next server to dispatch the mobile agent. Finally, S_1 computes $O_1 = Sig_{tPr1}(C_1, vh_1)$ and appends it with the preceding server's encapsulated offer and dispatches to the succeeding server S_2. This process of verification and computation will continue until the agent is back to its home with the required information or offer by the owner.

(c) **Agent returns to home S_0:** Finally, the agent returns to its home and gives the collected offers to its owner. The owner will recover all the encapsulated offers $O_0, O_1, O_2, O_3, \ldots O_{i-1}, O_i, O_{i+1}, \ldots O_n$ with the help of the temporary public key of every host. Where O_n is recovered by the tPb_n and then O_{n-1} is recovered with the help of the tPb_{n-1}, which is available in the hash function of the encapsulated offer O_n. After the verification of the integrity, the agent owner will recover the offers $o_0, o_1, o_2, o_3, \ldots o_{i-1}, o_i, o_{i+1}, \ldots o_n$ using the public key of the equivalent host and make use of it.

- **Multiple Colluded Truncation Attack prevention using 3-ID algorithm:** Xu et al (2006) identified the model that will protect the agent against two-colluder truncation attacks, and which will also avoid attacks by the single host [3]. And also they provided protection for multiple (more than two) colluded attacks while they are not in adjacent place. The advanced algorithm given here protects against the multiple colluded attacks, both in the adjacent and non-adjacent places because the chain relation is maintained in the form of the previous, current and then the next host identities. In the chain $S_0,\ldots,S_{i-1}, S_i, S_{i+1},\ldots, S_x, S_{x+1},\ldots, S_{m-1}$, S_m if S_i, S_{i+1}, and S_m are colluded to attack the sequence of offers from O_i, , O_{i+1}, …..… O_m. Then the offers will be O_0, O_1, , O_2, , O_3, … O_{i-1}, O_i`,$O_{i+1}$`,… O_m`. Now this chain is forwarded to the host S_{m+1} and it will recover the encapsulated offers and verify the identities using the hash function and easily identify the attack.

Fig: 1: Secure data Transfer in wireless Environment



4. Complexity Mitigation of 3-ID Algorithm

Mitigating the complexity is a major issue in developing the chain based verification model to protect the mobile agent data part. The proposed 3-ID verification algorithm also uses the chain verification concept to identify all types of attacks.

The complexity in this model is more because it verifies all the encapsulated offers right from the beginning. The complexity of the n^{th} remote host (S_n) to verify the offer is O(n).The remote host S_n will verify the encapsulated offers from its predecessors S_0 to S_{n-1}.

The total complexity to collect the information from N number of remote hosts by the agent is O(N(N+1)/2). This complexity is only for all the remote hosts not for the agent home. For example, the agent is dispatched from the owner to gather information from the three remote hosts. The remote host S_1 will verify the agent home offer and the remote host S_2 will verify the offer of agent home S_0 and S_1; again host three will verify the offer of S_0, S_1 and S_2. The total complexity is six but the verification complexity of home is not added as per the O(N(N+1)/2). If we add the verification complexity of agent home then it will be O(N(N+1)/2)+(N+*1*). This N+1 refers to the verification of the offer from host S_0 to host S_N.

To reduce the complexity of every host from O(N) to O($N/2$), the verification of the offers is reduced. The host S_n can verify the offers from S_n to $S_{n/2}$ instead of S_n to S_0. The verification and the computation time of the remote servers for both Go back N method and Go back N/2 method are given in Table 1. This result has is tested in the wireless Local Area Network (LAN) with the bandwidth of 54 mbps with 100 machines, where all the machines shares the data with one another. These two methods are implemented using Java in IBM Aglet [11]. Here the agent is forwarded by the administrator in the wireless network client host to gather the information about the list of process currently running. This information is mainly taken for intrusion detection, which is not discussed here clearly because it is out of scope of this paper.

Table 1 Comparison of N Verification and N /2 Verification

Servers	N **Verification**	N **/2 Verification**
S_1	305	305
S_2	375	344
S_3	430	375
S_4	476	375
S_5	523	422
S_6	562	430

Figure 2 shows the performance comparison of both the models. The Go back N/2 method takes less time than the Go back N method but there is no surety that the preceding host N/2-1 is genuine. To overcome this issue, we can apply the randomized check between the offers from host S_0 to host N /2-1.

Consider the scenario, where the multi-hop mobile agent is dispatched from its owner to gather information from the N (100) remote hosts. Now, the agent visited S_{n-1} (50) remote host and it is in the host S_n (51). The remote host S_n (51) will verify the offers from n-1 (50) to n - (n/2), i.e. 50 to 25(50-ceil(51/2) =51-26=25). The encapsulated offers from agent home 0 to remote server 24 can be verified randomly by hosts like 5 to 8 or 2 to 7, etc. Table 1 and Figure 2 show the time difference between N and N/2 verification. It shows that, with the N/2 complexity, the agent will return back to its home earlier than N complexity. It also reduces the agent hosts burden for verification of all the encapsulated offers from the preceding hosts.

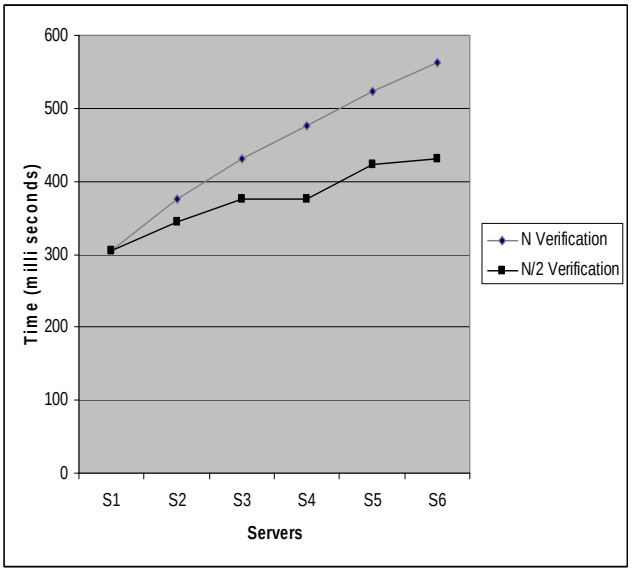


Figure 2 N verification Vs. N/2 verification

It is also possible to reduce the complexity from N/2 to N/3 or to N/4. In the total N remote servers, the 1 to N/4 remote servers can use N complexity and the N/4+1 to N/2 remote servers can use N/2 complexity and N/2+1 to (N/4+N/2) remote servers can use N/3 complexity and the (N/4+ N/2) to T remote servers can use the N/4 complexity. This type of reduced complexity in verification improves the smartness of the agent security environment.

5. Conclusion

Wireless environment is more vulnerable to the security issues. By incorporating this mobile agent concept, it will give more security issue in collecting the information from more than one host. To solve this issue, the 3-ID algorithm is an effective model to identify the attack on the information with the encapsulated three identities (current server identity, preceding server identity and succeeding server identity). The verification of the identity starts from the immediate proceeding host and terminates with the agent originator will

increase the verification complexity of the algorithm. To reduce the verification time complexity of the 3-ID algorithm, the *N* verification is reduced to *N*/2, *N*/3 and *N*/4 verification. With this extension, we can use it in the wireless network to gather any type of information with minimum cost and also with trusted level security.

Acknowledgement

Author would like to thank the originator of the 3-ID algorithm and also for his support in identifying the minimal time complexity model.

References

[1] Wong D., Paciorek N., Walsh T., DiCelie J., Young M. and Peet B. (1997), 'Concordia: An infrastructure for collaborating mobile agents', Rothermel K. and Popescu-Zeletin R. (Eds.), Proceedings of 1st International Workshop on Mobile Agents (MA'97), LNCS, Berlin, Germany, Vol. 1219, pp. 86-97.

[2] Yee B.S. (1997), 'A sanctuary for mobile agent', Technical Report CS97-537, UC San Diego, Department of Computer Science and Engineering.

[3] Karjoth G., Asokan N. and Gulcu C. (1998), 'Protecting the Computation Results of Free Roaming Agents', Proceedings of Second International Workshop on Mobile Agents (MA'98), Rothermel K. and Hohl F. (Eds.), LNCS, Vol. 1477, pp. 195-207.

[4] Hohl F. (1999), 'Mobile Agents and Active Networks', Proceedings of the IFIP Fifth International Conference on Intelligence in Networks (SMARTNET '99), Pathumthani, Thailand, pp. 45-51.

[5] Cheng J. and Wei V. (2002), 'Defenses against the truncation of computation results of free-roaming agents', Proceedings of 4th International Conference on Information and Communications Security, Lecture Notes in Computer Science, Vol. 2513, pp. 1-12.

[6] Zhou J., Onieva J. and Lopez J. (2004), 'Analysis of a Free Roaming Agent Result-Truncation Defense Scheme', Proceedings of 2004 IEEE Conference on Electronic Commerce, San Diego, USA, IEEE Computer Society Press, pp. 221-226.

[7] Zhou J., Onieva J. and Lopez J. (2004), 'Protecting Free Roaming Agents against Result-Truncation Attack', Proceedings of 60th IEEE Vehicular Technology Conference, Los Angles, USA, pp. 3271-3274.

[8] Xu D., Harn L., Narasimhan M. and Luo J. (2006), 'An Improved Free-Roaming Mobile Agent Security Protocol against Colluded Truncation Attacks', Proceedings of the 30th Annual international Computer Software and Applications Conference(COMPSAC'06), Vol. 2, pp. 309-314.

[9] Venkatesan S. and Chellappan C. (2007), 'Protecting Free Roaming Mobile Agent against Multiple Colluded Truncation Attacks', ACM Proceedings of International Conference on Mobile Multimedia Communication(MobiMedia'07), Greece, pp. 293-297.

[10] Venkatesan S. and Chellappan C. (2009), 'Free Roaming Mobile Agent (FRoMA) Protection against Multiple Attacks', International Journal on Communication Networks and Distributed Systems, Vol. 3, No. 4, pp. 362-383.

[11] Aglet (2004), http://www.aglets.sourceforge.net/.

Ashish Kumar Srivastava is pursuing Ph.D.(IT) from MANIT Bhopal in Security Issues in Wireless adhoc Network. He has received his M.Tech(IT) degree from Tezpur Central University Tezpur Assam, India. In year 2002 and his B.Tech degree in (EEE) in the year 2000.His research interest area is Security in infrastructure less Networks, Information Security, and Wireless sensor network.

Dr. Aditya Goel did his B.E. (Hons) in Electronics & Communication and M.Tech in Communication Engineering from I.I.T. Bombay. Subsequently he was awarded with the Ph.D. degree in Electronics & Communication Engineering in the Year 2000. He has teaching & research experience of more than 20 years. Presently he is working as Associate Professor in the Deptt. of Electronics & Communication Engineering at Maulana Azad National Institute of Technology(Deemed University) , Bhopal, India. He has published more than 70 Research papers in reputed International Journals & Conferences including IEEE. His areas of interests are Optical Communication, Digital Signal Processing, Optical Networks, Computer Communication etc . He has guided several M.Tech dissertations and Ph.D thesis on these areas. He had been Principal Investigator of the research project titled " Broadband Lightwave communication systems" sanctioned by Govt. of India, New Delhi. Presently also he is working as Principal Investigator of the R&D Project " High speed optical components for WDM systems" sanctioned by M.H.R.D. Govt. of India , New Delhi. He has reviewed many research papers of International conferences & journals including SPIE. He is a Life Fellow member of the Institution of Electronics & Telecommunication Engineers and Senior Member of Computer Society of India.