I.J. Wireless and Microwave Technologies, **2011, 2, 70-74**
Published Online April 2011 in MECS (http://www.mecs-press.net)
DOI: 10.5815/ijwmt.2011.02.10

Available online at http://www.mecs-press.net/ijwmt

A Security Routing Protocol Protecting Mobile Agent Against Cluster Attack

Wenbing Wang, Zhifeng Zhang

School of Computer and Communication Engineering, Zhengzhou University of Light Industry, 450002, Henan Zhengzhou, China

Abstract

The security issue of MA(mobile agent) is concerned for a long time, Especially for attack from malicious hosts. The paper proposes a new security routing protocol whose highlight is integrating group signature into MA system. It will decrease the probability that malicious hosts cluster to skip one host deliberately, and avoid too much invalid connection to host who is offline or rejective to service. Besides, in new protocol, each host will hold less public keys of other hosts and save time on verifying MA. The last superiority of new protocol is improving the efficiency of *DoS*(denial of service) attacker tracing.

Index Terms: Component; mobile agent; cluster attack; group signature

1. Introduction

MA(mobile agent) is an executable program which can roam on network and complete the mission assigned by MA owner. Compare to C/S model, MA has more advantage on network management, fault tolerance and scalability. It is an extension and substitution for traditional C/S model, and will be significantly helpful to distributed computing, electronic business, intrusion detection, software distributing, information collection and network management.

Nevertheless, MA system has a variety of security issue that comes down to two sides: Threaten to visited hosts from MA and threaten to MA from visited host. Today, the first one has gained great development, like protecting visited hosts by authentication and trust mechanism[1]. As for the second one, development is still at preliminary stage. Reference [2] offers a scheme that utilizes encryption function to prevent executable code from modification. Reference [3] proposes a security protocol which is designed to protect code and status of MA against visited host's breach. However, so far, there is no available way to avoid deleting MA since visited hosts have chance to interrupt MA's execution after they attain the MA.

* Corresponding author.
E-mail address: wang_wb1978@yahoo.com.cn

The paper defines DoS attack that MA cannot return MA owner on time due to MA deleting by visited hosts. Apparently, DoS attack happens unexpectedly and results in tremendous shock. The protocol in [4] accurately locates malicious host after attack but is at the end of its resource on other problems like group attack. In the course of our due diligence, we put forward introducing group signature to MA system for reinforcing the protocol in [4]. Group signature advanced by Chaum[5] arouses extensive attention. It has five core principles: any member in group can sign on behalf of group; validation is completed with unique group public key; anonymity of group signature is cancelled by specified group authority; its attributions include: correction, anonymity, unlinkability, unclusterability and traceability. Group signature's nature meets with security request for MA routing protocol.

2. The problems in routing protocol proposed by [4]

By research, we consider that the routing protocol proposed in [4] has following defects on detail.

- Every visited host has right to independently choose the next host to pass MA on, as a result, MA owner is not able to find the real reason why some hosts did not get MA. Some hosts may take advantage of this and collaborate for letting MA skip certain host.
- Protocol proscribe that the number of connecting to each host is limited to *m* for preventing certain host from unbounded connection. In the process of protocol, sequence of visited hosts changes dynamically, so it is hard to restrict the maximum number of connection to hosts. The algorithm *SelectNextHost* in [1] did not solve the problem too.
- Each host needs a completed visited host list to choose unvisited hosts as next MA receiver. The representation of visited host list is: $vc^{\#(cj)}=vc^{\#(ck)};c_j;sig_{Cj}(vc^{\#(ck)};c_j)$(now assuming that MA migrates from c_k to c_j). For getting the plain format of visited host list, each host has to own public keys of all the other hosts. If there are large numbers of hosts involved, each host has to spend too much time on decode *vc* list.

3. The security routing protocol basing on group signature

If the group involved in group signature is dynamic, many unexpected new elements and security issues are produced. Dynamic group model, correlative security definition and proved secure system framework described in [6] is Logical proofs with which the paper introduces group signature into MA routing protocol. The first step of new protocol is that the visited hosts are divided into groups, and number of each group members over m is not allowed. m is a system parameter whose value is relevant to quantity of visited hosts.

The new representation of MA changes to: agenti,j=(bc；mdi,j；uid；r；vc#(ci,j)). The meaning of variables bc, mdi,j and uid keep same. In the representation of MA, r is(r，sigh(r)). If MA owner divides the visited hosts into N groups, each group include mi hosts(0<i<=N) and r is {{c1,1,c1,2…c1,m1},{ c2,1,c2,2…c2,m2},….{ cN,1,cN,2…cN,mN}}.after executing MA, the ith host in jth group-ci,j-will create a visited hosts list whose format is {sigg1(g1,m1#), sigg2(g2,m2#)…siggi-1(gi-1,m(i-1)#),gi,j}. sigg1(g1,m1#), sigg2(g2,m2#) and siggi-1(gi-1,m(i-1)#) means that the last visited host of these three groups sign on their visit hosts list in the name of their group. Assuming that the jth member of the ith group gets MA from the kth member in the same group, we define the format of gi,j is {gi,k,ci,j,sigci,j(gi,k,ci,j)},of which gi,k and gi,j has uniform encryption method; If ci,j is not in the same group with receiver who could be in the i+1th group or MA owner, on behalf of the ith group, ci,j sign on gi,j to create siggi(gi,j) as the last element of vc#(ci,j).This explains how sigg1(g1,m1#), sigg2(g2,m2#) and siggi-1(gi-1,mi-1#) come. Here, #(ci,j) is the total number of hosts in vc#(ci,j).

SelectNextHost is an algorithm executed on sender that is used to select next host for passing MA on. In the arithmetic, MA owner firstly makes certain the hosts that MA owner wants MA visit. Then it divides them into

This work was supported by a grant from Henan province key science and technology project(No. 102102210544)

GROUPNUMBER groups and each group has Mgid members. Assert in(e,s) is to distinguish whether a certain element e is in set s. for transfer MA, each sender has buffer buf to save IP address of the offline hosts and the hosts that didn't return valid confirmation. Function Online is used to test whether a host is online. Operator ¬ on binary number is to get the number's complement. k is the global variable whose initial value is 1. If ci,j is the last visited host of its group and the ith group is the last group, MA will return MA owner directly; if ci,j is the last visited host of its group but the ith group is not the last group, ci,j choose someone of next group-i+1th group-as receiver; if ci,j is not the last visited host of its group and the ith group is not the last group, ci,j will choose a receiver who is not in vc#(ci,j) and its buffer buf. The following is algorithm SelectNextHost.

```
while(k<=mi)
    if(k==j)
        k=k+1
    else
        if(¬in(c_{i,k}, vc^{#(ci,j)}) & ¬in(c_{i,k}, buf))
            append c_{i,k} to buf
            if(Online(c_{i,k}))
                NextHost= c_{i,k}
                k=mi+2
            else
                k=k+1
            endif
        else
            k=k+1
        endif
    endif
endwhile
if(k==mi+1)
    if(i==GROUPNUMBER)
        NextHost=h
    else
        s=1
        while(s<=m_{i+1})
            if(¬in(c_{i+l,s}, buf))
                append c_{i+l,s} to buf
                if(Online(c_{i+l,s}))
                    NextHost= c_{i+l,s}
                    s=m_{i+1}+2
```

```
                    else
                        s=s+1
                    endif
                else
                    s=s+l
                endif
            endwhile
        endif
    endif
```

Sending protocol

- *SelectNextHost* is used to choose next host(maybe it is not in same group with sender)or MA owner as MA receiver.
- Basing on whether the receiver is in same group with itself, MA sender creates specify *vc* list.
- Save MA to local database.
- Send MA to the host chosen by algorithm *SelectNextHost*.
- Wait for confirmation from receiver. If get valid confirmation in certain time span, skip to step 6; or return to the first stip.
- Save confirmation and identification of receiver(sender and receiver are in same group)or the receiver's group(sender and receiver are not in same group).
- Delete MA from local database.

Receiving protocol

- Receive MA.
- Verify each augment of MA. If they are valid, go to next step; otherwise, return MA to MA owner and exit from receiving process.
- Create confirmation for sender.
- Return confirmation to sender.

In above protocol, each receiver need to return sender a confirmation which's format depends on whether receiver and sender in the same group. If MA transfers from ci,k to ci,j, the confirmation presented by ci,k has to signed by itself; if MA transfers from ci,k to ci+1,1(the first member of next group in vc), the confirmation presented by ci+1,1 need be signed by ci+1,1 representing its group.

The last but not the least part of protocol is modifying tracing process. If MA don't return MA owner on time, MA owner start a tracing process to find out the malicious accountable host. First, MA owner sends to the first group's authority a tracing request which includes confirmation received from a member of first group. According to the characteristic of group signature, group authority is capable of allocate the member who signed confirmation. If the signer has illegally passed MA, it can present a confirmation received from its next member in vc. The process described above is repeated until MA owner finds out a group member who can't present a valid confirmation. This group member is the host who carried on DoS attack.

4. The capability enhanced by new protocol

- 1) When executing *SeleetNextHost*, sender has to check every host's qualification for choosing receiver. Therefore, all the hosts except sender in the group must collaborate to form a cluster for skipping one host. In new protocol, the rule that the most trusted host has competency for group authority adds difficulty for clustering.
- 2) New protocol defines that the sum of members in each group over *m* is not allowed. So, for the host who denies service, its repetitions of being connected are less than *m*. The improvement decreases the number of unnecessary connecting.
- 3) For selecting receiver, each sender must know the clear text of its group's visited hosts list. In new protocol, if sender and receiver are in the same group, the expression of visited hosts list is as following: $vc^{\#(ci,j)}=\{sig_{g1}(g_{1,m1\#}),sig_{g2}(g_{2,m2\#})\ldots sig_{gi-1}(g_{i-1,mi-1\#}),g_{i,j}\}$. the succeeder of $c_{i,j}$ only need use public keys of other hosts in the same group to decode $g_{i,j}$. If receiver and sender are not in same group, the expression of visited hosts list created by $c_{i,j}$ is as following: $vc^{\#(ci,j)}=\{\mathrm{sig}_{g1}(g_{1,m1\#}),\ \mathrm{sig}_{g2}(g_{2,m2\#})\ldots\mathrm{sig}_{gi-1}(g_{i-1,mi-1\#}),\mathrm{sig}_{gi}(g_{i,mi\#})\}$, and receiver decodes $\mathrm{sig}_{gi}(g_{i,mi\#})$ by the *ith* group's public key.

5. Conclusion

The paper pointed out disadvantages of security routing protocol in [1], and designed a new protocol basing on group signature to make up deficiency and enhance efficiency, security and tightness. In the end, the paper analyzes the cause that new protocol is more advanced than the old one and is definitely effective for MA application.

Acknowledgment

During writing the paper, we get a lot of help from Weidong Qiu, my supervisor in Shanghai Jiaotong University. Here I thank him sincerely.

References

[1] Jansen W A, "A privilege management scheme for mobile agent systems," First International Workshop on Security of Mobile Multi-Agent Systems. SEMAS-2001.

[2] Tomas Sander, Christian F Tschudin, "Protecting mobile agents against malicious hosts," Lecture Notes in Computer Science.1998, 1419 : 44—49.

[3] Hohl F, "Time limited blackbox security : protecting mobile agents from malicious hosts," Lecture Notes in Computer Science. 1998, 1419 : 92—93.

[4] Cubaleska Biljana, Qiu Weidong, Schneider Markus, "How to play sherlock holmes in the world of mobile agent," Lecture Notes in Computer Science. 2002, 2384 : 449—454.

[5] Chaum D, van Heyst E, "Group signatures," Lecture Notes in Computer science. 1991, 547 : 257—265.

[6] Bellare Mihir, Shi Haixia, zhang Chong, "Foudations of group signatures : the case of dynamic groups," Lecture Notes in Computer Science. 2005, 3376 : 136