Advanced Engineering Forum *Online: 2012-09-26*
ISSN: 2234-991X, Vols. 6-7, pp 913-918
doi:10.4028/www.scientific.net/AEF.6-7.913

A Novel Anonymous Authentication Scheme in ad Hoc Networks

Jizhi Wang, Yinglong Wang, Shujiang Xu

(Shandong Provincial Key Laboratory of Computer Networks,

Shandong Computer Science Center, Jinan, China, 250101)

Keywords: ad hoc network, authentication, anonymity, discrete logarithms problem

Abstract: Many authentication protocols in ad hoc networks are identity based, which means that in order for one node to trust another, it needs to know the other node's identity. Hence there exists an inherent trade-off between trust and anonymity. We propose an anonymous authentication scheme, where each node, instead of using its real identity, generates a dynamic pseudonym using a one-way hash function. With the help of a CA offline, the scheme can realize the authentication procedure based on discrete logarithms problem. We analyze the security and the anonymity in the scheme, which shows that it is very valid.

Introduction

As a model of communication, the routing in ad hoc networks has recently gained much attention [1-3]. Query flooding is the most popular routing request method used in such networks. One drawback is the fact that it might compromise users' privacy. The IP addresses of requesters and responders can easily be discovered. Hence, many studies focus on providing anonymous routing in such networks [4-5].

On the other hand, numerous concerns have been raised about the issue of providing authentic resources in the systems. To guarantee that real resources are received from authentic responders, some researchers have built trust models to help nodes verify the validity of other entities [6-8]. Most trust models, however, are identity based, which means that for one node to trust another, it needs to know the identity of the other node. Thus, there exists an inherent trade-off between trust and anonymity.

The purpose of designing an anonymous authentication protocol in ad hoc networks is motivated by a specific problem: how to support authentication without exposing the real identities of nodes. In this paper, we propose the design of the scheme, in which each node generates a dynamic pseudonym using a one-way hash function. Such one-way mapping can effectively defend against impersonation and forgery, so that the dynamic pseudonyms can be used as the real IDs in ad hoc networks. We also design a novel authentication scheme based on discrete logarithms problem to help unfamiliar nodes successfully complete authentication procedures during transactions. The salient features of the scheme include the following:

1. Achieving anonymity for authentication. The scheme enables dynamic-pseudonym-based trust management so that the real identities of nodes are protected during the authentication. The scheme can be adopted for anonymizing communications.

2. Non-repudiation. Maybe a node does not admit having sent some messages because its ID is pseudo ID. Hence, non-repudiation is very important for anonymous protocols. Our scheme can get non-repudiation.

3. Resisting impersonation. An attacker can arbitrarily intercept, modify, or forge the message between two authentication parties so that the attacker can impersonate either of them to another party. The scheme, however, employs the one-way hash function to bind users' pseudonyms. Using the one-way hash function, users in the scheme authenticate the opposite party based on discrete logarithms problem. Our theoretical analysis shows the security of the scheme under the attack.

The rest of this paper is organized as follows: Section 2 introduces related works including trust management scheme and anonymous ad hoc networks protocols. Section 3 presents the scheme design. Section 4 analyzes the security of the scheme. Section 5 concludes this work.

Related works

In this section, we briefly describe related works in authentication and anonymity.

Authentication schemes. In literature [6], an authentication scheme based on zero-knowledge proof was presented. The authentication scheme includes two phases: network layer operations and data link layer operations. In the network layer, the node authentication procedure attempts to authenticate the true identity of the communicating nodes through a non-iterative zero-knowledge protocol. Likewise, in the data link layer, a challenge-response protocol is used in authentication procedure. Thus the two phases enhance the security of ad hoc networks.

In literature [7], an authentication architecture for collaboration among agents is proposed in ad hoc networks without security assurance. The architecture requires that there should exist at least one secure node. The secure node generate authentication codes using random numbers and agent information, and distributes the codes among the agents. Based on the distributed random value and the authentication code, the agents can be authenticated. In the architecture the random number and the authentication code are publicized information.

In literature [8], a two-tier authentication scheme for cluster and individual sets. The first tier, based on a hash function and the MAC concept, provide fast message verification and group identification. The second tier, based on secret sharing technology, provides secure user identification. The scheme can prevent internal and external attacks, including black holes, impersonation, routing table overflows and energy consummation attacks.

Generally, many trust design are identity based, where one node does not trust another before knowing its identity.

Anonymous routing. Privacy has become an increasingly salient issue, and considerable progress has been made with anonymous communications. Several solutions achieve mutual anonymity for both initiator and responders in ad hoc networks [4-5], which generally aim at concealing the real identities of users during transactions. In MASK [4], nodes construct an anonymous path based on the secret-sharing technique, providing complete and mutual anonymity for nodes. In literature [5], a distributed routing protocol is proposed by encrypting routing packet header and abstaining from using unreliable intermediate node.

Anonymous authentication. To the best of our knowledge, the research on anonymous authentication in ad hoc network is seldom. In the literature [9], an anonymous channel and authentication is proposed in GSM networks. The scheme based on blind signature generates anonymous channel, and then uses pseudo ID to realize authentication. In the literature [10], a zero-knowledge authentication scheme called Pseudo Trust (PT) is proposed in P2P networks. We will analysis the scheme because P2P networks are similar with ad hoc networks.

In [10], each node randomly chooses two large primes p_1 and p_2 and calculates the integer $n=p_1 \times p_2$. Then the node adopts a hash function to generate a seed=h(ID, p_1,p_2) . The pseudo ID (PI) and pseudo ID certification (PIC) can be computed by the node using the seed. The PI and PIC of nodes are published on some well-known Websites. When two nodes need to authenticate each other, they rely on a zero-knowledge proof to finish authentication procedure.

Though analysis for the scheme, we find that there are two drawbacks in the scheme. The detailed describes are as follows.

1) Static pseudonimity. Each node in PT can generate a pseudo ID that are not changed during the node's lifetime. If an eavesdropper can receive two messages with the same pseudo ID, he/she can ascertain that two messages are from the same node. He/she may know who is the holder with the pseudo ID when he/she collects enough messages with a same pseudo ID. Hence, PT can only provide pseudonimity rather than anonymity, i.e. pseudonimity is not equal anonymity [11]. Because pseudo ID and real ID are also string, they have no difference if someone can use a pseudo ID to access its holder. Hence, using a static pseudo ID can not obtain anonymity.

2) Pseudo identity security in unsafe public sites. In PT, the pseudo identity and pseudo identity certificate of each node are published on some well-known websites. As mentioned in [10], if a powerful adversary cracks these websites, he/she would change some PI's moduli n to another one of which he/she holds the factors, which PT can resistant. However, in fact, the powerful adversary

needn't change a pseudo ID that have existed in the website. He/she can generate a new PI and PIC and add it in the website. Other nodes can not distinguish the new PI and PIC from others because all of the IDs are pseudo. Thus, the adversary can use his PI and PIC to be authenticated by other nodes.

Considering the two problems, we use dynamic pseudonimity with the help of an offline CA to solve them.

The scheme

The real and specific challenge that underlies the trade-off between trust and anonymity is that, on the one hand, all existing trust systems attempt to link each node ID with a trust value; on the other hand, anonymous designs hide the real IDs of communicating parties during transactions. Instead of using real IDs to deal with other nodes in an ad hoc network, can nodes use pseudonyms to interact with others and accumulate their reputations? Clearly, if we would like to adopt such a mechanism, we need to guarantee that when a node selects a pseudonym, it is not likely to be a name already being used by another node; and that pseudonym impersonations must be made impossible.

In this section, we first give an assumption of the design of the scheme and then discuss detailed authentication procedure.

Assumption. First, four conditions are given before the scheme can be applied.

1) Before our anonymous authentication protocol runs, it is supposed that nodes have built anonymous link via anonymous routing, such onion routing.

2) In ad hoc networks, there is a CA.

3) Each node has been authenticated by the CA before it joins the network. Thus, each node has a private key and a public key certificate signed by CA.

4) Each node authenticated by CA is honest.

Authentication process. Because there is a CA in networks, there are two cases that one is to use CA during authentication process and the other is not to use CA during authentication process.

With a CA during authentication process

If we can utilize a CA during authentication process, designing an anonymous authentication protocol is very simple.

Step 1: Node I generate a pseudo identity and sign it with its private key, and, then, send it to the CA.

Step 2: The CA verifies the I's signature using I's public key. After successful verification, CA signs the I's PI using its private key, and, then, sends it to Node I.

Step 3: Node I send its PI signed by CA to Node R. Then, Node R verifies the CA's signature using CA's public key.

Then, Node R runs the same steps. Thus, Node I and R can realize mutual anonymous authentication.

However, the above authentication protocol is not suitable for ad hoc networks because Node I and R need to communicate with CA frequently during authentication process. Hence, we consider a novel anonymous authentication protocol without a CA during authentication process.

Without a CA during authentication process. First, terms are defined as follows.

Table 1: Notations of Variables

notation	specification
I	Initiator of a query
R	Responder
M	Malicious node
PI_A	The pseudo identity of node A
ID_A	The real identity of node A
$Cert_A$	Public key certificate of node A
$Sign_A\{\}$	Signature of node A
h	Hash function
G	Finite cyclic group
g	The generator of G
p	The element number of G

As mentioned in Section 3.1, before joining the network, a node must authenticated by a CA via offline mode. So the node has a private key, a public key and a real identity that is connected with the public key. Then we design the anonymous authentication protocol based on discrete logarithm problem.

At first, CA selects a finite cyclic group G with p elements. Let g be a generator of G, then CA publishes g and p. Then, CA selects a secret random number $s \in \{0,1,\ldots,p-1\}$ and sends s encrypted with nodes' public key to each node authenticated. Thus, CA and all nodes share a secret number s. When a new node joins the network or an old node leaves the network, the secret number s should be updated.

Then, we can see how to realize anonymous authentication without a CA during authentication procedure.

When the query initiator, node I initiates an authentication procedure, suppose R is the responder. Node I sends an authentication request to R through the anonymous path. R sends a challenge message and verifies the node I. Then I verifies that R is a legitimate node. The detailed procedure is as follows.

1. PI Generation: Node I randomly chooses a number $a \in \{0,1,\ldots,p-1\}$. Then, it use its private key to sign $\{ID_R, Cert_I, s, a\}$. Use h to generate a PI_I, where $PI_I = h(ID_I, Cert_I, s, a, Sign_I\{ID_I, Cert_I, s, a\})$.
2. Authentication request: Before starting authentication, node I computes u=h(PI, a). Then, I sends {PI, a, u} to node R.
3. Request verification: Node R computes u'=h(PI, a), then verifies whether u=u'. If the verification holds, node R goes on to the next authentication step, otherwise it rejects this authentication request.
4. PI Generation: Node R randomly chooses a number $b \in \{0,1,\ldots,p-1\}$. Then, it use its private key to sign $\{ID_R, Cert_R, s, b\}$. Use h to generate a PI_R, where $PI_R = h(ID_R, Cert_R, s, b, Sign_I\{ID_R, Cert_I, s, b\})$.
5. Challenge: node R computes $v=h(PI_R, b)$. Then, R sends $\{PI_R, b, v\}$ to node I.
6. Proof generation: Node I sends the following to node R:

$x=(g^b)^s \bmod p$

7. Verification: Node R checks

$x=(g^s)^b \bmod p$

8. If the verification holds, node R sends the following to node I

$x=(g^a)^s \bmod p$

9. Then, Node I checks

$x=(g^s)^a \bmod p$

If the verification holds, node I and R finish the mutual anonymous authentication.

The security analysis

We first analyze the anonymity and security of the scheme and, then, discuss the attack resilience.

Anonymity. Our scheme generates dynamic pseudo IDs from real identities using a cryptographic hash function. Because a node uses different random number a or b to run authentication protocol, other nodes can not judge which node these messages should be connected with.

The anonymity of a node's identity comes directly from the one-way property of cryptographic hash function. Let h() be a hash function with m-bit-long hash values and assume it is well designed and has no structural drawback for cryptanalysis. In cryptograph terminology, h() takes advantage of a pre-image-resistance property, i.e., for any given hash value y, it is computationally infeasible to find an x such that h(x)=y.

A malicious node may launch advanced attacks, such as finding two different but real identities so that the two identities have the same PI. It might then use one of the two identities to impersonate the node with the other identity. However, this kind of attack is withstood by the collision-resistance of hash function.

The security of anonymity. Because our scheme uses pseudo ID to authenticate, the two parties of communication do not know real ID each other. Could a malicious node use a pseudo ID to communicate with other nodes? Obviously, it can not both because it does not know the secret number s and it can not extract s from previous messages because of discrete logarithms problem.

Non-Repudiation. Because nodes use dynamic pseudo ID to authenticate, no one can connect a pseudo ID with an entity. Maybe a dishonest node does not admit having sent a message. However, our scheme can resistant this attack.

Supposing that node I does not admit having sent a authentication message, the opposite node R send its PI and random number a to CA. Then, CA asks I to send $Sign_I\{ID_I, Cert_I, s, a\}$ and ID_I. Firstly, CA verifies the signature using I's public key. Secondly, CA verifies I's PI using hash function. Thus node I can not deny having sent messages.

The leakage of secret number. Because the secret number s is a key in the scheme, it will be a serious problem if the secret number is leaked. Hence, in the scheme, CA will update the secret number s when a new node joins the network or an old node leaves the network so that new nodes do not know previous secret number and old nodes do not know next secret number. Hence, CA issues the new secret number encrypted using each node's public key and send to each node. Then, each node decrypts it using its private key and obtains new secret number. Thus, it can be sure that the secret number is not leaked.

Impersonation. The scheme can effectively defend against impersonation. If our scheme executed successfully between the initiator I and responder R, the secret number s is not leaked and any adversary node can not compute s through collecting enough messages. Without the secret number, a malicious node can not impersonate others. We prove this feature below.

Theorem 1. Assume it is computationally infeasible for discrete logarithms problem, suppose a malicious M interacts the scheme with responder R to impersonate initiator I and convince R that it is I. Then, the probability that M succeeds is 1/p.

Proof. According to the authentication procedure steps in Section 3, M must know the secret number s so that it can impersonate initiator I. Because it is computationally infeasible for discrete logarithms problem, M can not compute s. Thus, M can guess a secret number s', then, computing $x=(g^b)^{s'} \bmod p$. When s'=s, R can verify x. Hence, the probability of soundness that M guesses is 1/p, i.e. the probability that M succeeds is 1/p.

Likewise, a malicious node can not impersonate the responder R.

Denial of service attack. DoS attack has gained increasing concern in distributed networks. Typically, this attack renders networks, hosts, and other victim systems unusable by consuming the bandwidth of victim networks or deluging them with a huge number of requests to overload their systems. In ad hoc networks, the system may suffer from DoS attacks during authentication interactions. Since I choose a desired responder to start an authentication procedure, I does not suffer from DoS attacks. Thus, we simply focus on the possibility that responders are under DoS attacks. In the scheme, R asks I to verify first, and, then, I asks R to verify. If we transpose this sequence, attackers may send a lot of authentication requests to the targeted R with forged PIs. R has to answer each request by generating a random number and sending it back. R then waits for challenges from these nodes, which will never come. Since R only need to generate random numbers, a DoS attack hardly has an effective influence on R. Thus, the scheme can effectively defend against DoS attack.

Conclusions

Due to the inherent trade-off between trust and anonymity, identity-based trust management schemes cannot be directly employed in anonymous ad hoc networks. We propose an anonymous authentication protocol. In this work, an authentication scheme is designed to support trust management in anonymous environments, so that nodes may use dynamic pseudonyms instead of their real identities in ad hoc networks.

We prove that the probability of a successful impersonation is computationally infeasible, even if the adversaries have collected all of the previous authentication messages. We believe that wide deployment of this design will provide better privacy and security for ad hoc networks.

Acknowledgement

This work was supported by the Natural Science Foundation of Shandong Province, China (No.ZR2011FM023), and Natural Science Foundation of China (No.61070163, No.60973146)

Reference

[1] Lenders V., May M., Plattner B.. Density-based anycast: a robust routing strategy for wireless ad hoc networks. IEEE/ACM Transactions on Networking, 2008, 16(4) 852-863

[2] Xu Jia, Li Zhi, Li Qianmu, Liu Fengyu. An adaptive clustering routing transition protocol in ad hoc networks. Computer Communications, 2008, 31(10) 1952-1960

[3] Ahmad AI Hanbali, Philippe Nain, Eitan Altman. Performance of ad hoc networks with two-hop relay routing and limited packet lifetime. Performance Evaluation, 2008 65(6) 463-483

[4] Yanchao Zhang, Wei Liu, Wenjing Lou, Yuguang Fang. MASK: anonymous on-demand routing in mobile ad hoc networks. IEEE Transactions on Wireless Communications, 2006, 5(9) 2376-2385

[5] Azzedine Boukerche, Khalil EI-Khatib, Li Xu, Larry Korba. An efficient secure distributed anonymous routing protocol for mobile and wireless ad hoc networks. Computer Communications, 2005, 28(10) 1193-1203

[6] Komninos N., Vergados DD., Douligeris C.. Multifold node authentication in mobile ad hoc networks. International Journal of Communication Systems, 2007, 20(12) 1391-1406

[7] Yasukuni Okataku, Nobukazu Yoshioka, Shinichi Honiden. An authentication architecture for collaboration among agents in ad hoc networks. Electronics and Communication in Japan, 2004, 87(5) 11-19

[8] Yuh-Ren Tsai, Shiuh-Jeng Wang. Two-tier authentication for cluster and individual sets in mobile ad hoc networks. Computer Networks, 2007, 51(3) 883-990

[9] W. S. Juang, C. L. Lei, C.Y. Chang. Anonymous channel and authentication in wireless communication. Computer Communications, 1999, 22(15) 1502-1511

[10] Lu L., Han JS., Liu YH., Hu L., Huai JP., Ni LM., Ma J.. Pseudo trust: zero-knowledge authentication in anonymous P2Ps. IEEE Transactions on Parallel and Distributed Systems, 2008, 19(10) 1325-1337

[11] Kavakli, E., Kalloniatis, C. Gritzalis S.. Addressing privacy: matching user requirements to implementation techniques. The 7th Hellenic European Research on Computer Mathematics & its Applications Conference, 2005, Athens, Greece.