2006

Authenticated AODV Routing Protocol using one-time signature and transitive signature schemes

Shidi Xu
University of Wollongong, sdx86@uowmail.edu.au

Yi Mu
University of Wollongong, ymu@uow.edu.au

Willy Susilo
University of Wollongong, wsusilo@uow.edu.au

Authenticated AODV Routing Protocol using one-time signature and transitive signature schemes

Abstract

Mobile ad hoc network (MANET) has been generally regarded as an ideal network model for group communications because of its specialty of instant establishment. However, the security of MANET is still a challenge issue. Although there are some existing security schemes such as ARAN (Authenticated Routing for Ad hoc Networks) protocol that makes use of cryptographic certificate to provide end-to-end authentication during routing phases, the overhead of security computation is still a serious hurdle for real application. In this paper, we propose a comparatively efficient scheme to perform ARAN protocol, based on AODV, by using one-time signature in place of conventional signature, aiming at achieving the same level of security but improved efficiency. We also provide two approaches to handle the authentication of gratuitous route reply using delegation token and transitive signature schemes.

Disciplines

Physical Sciences and Mathematics

Publication Details

Xu, S., Mu, Y. & Susilo, W. (2006). Authenticated AODV Routing Protocol using one-time signature and transitive signature schemes. Journal of Networks, 1 (1), 47-53.

Authenticated AODV Routing Protocol Using One-Time Signature and Transitive Signature Schemes

Shidi Xu
University of Wollongong, Wollongong, Australia
Email: sdx86@uow.edu.au

Yi Mu and Willy Susilo
University of Wollongong, Wollongong, Australia
Email: { ymu, wsusilo }@uow.edu.au

***Abstract*—Mobile ad hoc network (MANET) has been generally regarded as an ideal network model for group communications because of its specialty of instant establishment. However, the security of MANET is still a challenge issue. Although there are some existing security schemes such as ARAN (Authenticated Routing for Ad hoc Networks) protocol that makes use of cryptographic certificate to provide end-to-end authentication during routing phases, the overhead of security computation is still a serious hurdle for real application. In this paper, we propose a comparatively efficient scheme to perform ARAN protocol, based on AODV, by using one-time signature in place of conventional signature, aiming at achieving the same level of security but improved efficiency. We also provide two approaches to handle the authentication of gratuitous route reply using delegation token and transitive signature schemes.**

***Index Terms*—MANET, Routing, AODV, Digital Signature, One-time signature, Transitive signature.**

I. Introduction

The Mobile Ad hoc Networks (MANET) are a specific type of network. Just as its name implies, it is formed by mobile nodes, such as laptops and PDAs. The construction of the networks is generally impromptu, therefore, networks can be formed whenever required and topology is changing from time to time. Ideally, any nodes satisfy general entering conditions will be accepted as a legitimate member of the network. These properties make MANET very suitable for group communications, in which, a number of people get together, forming a network to share documents and exchange conversations.

On the other hand, the wide-open environment makes this network super vulnerable to inside and outside attacks [1**Error! Reference source not found.**]. Especially in the case of routing [2], since the absence of central control, it is extremely difficult to prevent nodes from behaving improperly. Although there exist a large number of MANET routing protocols [3,4,5, 8,11], most of them were designed without any security considerations (generally it is assumed that all nodes are friendly). Besides, the resource constraints (both computation and bandwidth) of MANET put up great difficulties over the deployment of security. Two widely known reactive routing protocols are AODV (Ad hoc On-Demand Distance Vector Routing) [8] and DSR (Dynamic Source Routing) [5], which are both very efficient but are subject to a variety of attacks.

To reinforce the security of routing, ARAN [11] makes use of cryptographic techniques to offer security in an open-manage environment. Since the security is based on public key cryptography, the efficiency of ARAN is under suspicion. In this paper, we pursue the advantages of one-time signature, which is more efficient in signing and verification, to replace conventional digital signature in protecting routing packets, though, at the same time, maintaining the same level authentication.

In our previous work [12], we made use of delegation token to enable the authentication of the gratuitous reply in route discovery. In this paper, we introduce another approach by using transitive signature scheme introduced by Micali and Rivest [6].

The rest of the paper is organized as below. Section 2 briefly introduces the AODV routing protocol and ARAN routing scheme. Section 3 describes the HORS one-time signature scheme and its key generation process. Section 4 explains our scheme used to secure AODV, called authenticated AODV. In section 5, we introduce two approaches to be used to authenticate gratuitous route reply. Section 6 discusses the security of our proposal. The last section concludes the paper.

II. Backgrounds

In this section, we introduce the basics of the AODV routing protocol and the ARAN authentication scheme.

A. AODV Routing

AODV is a simple and efficient on-demand ad hoc routing protocol. Basically, it uses RREQ (route request), RREP (route reply) and RRER (routsue error) messages to accomplish route discovery and maintenance

operations. It also utilizes sequence numbers to prevent routing loops. Routing decision making is based on sequence numbers and routes maintained in each node's routing table.

The routing operations of AODV generally consist of two phases: route discovery and route maintenance. Route discovery is performed through broadcasting *RREQ* message. Whenever a node needs to send data packets to a destination, it first checks if it has an existing route in the routing table. If not, the source node will initiate a *RREQ* and broadcast this request to all the neighbours. Then neighbouring nodes will update their routing table according to the received message.

When *RREQ* reaches the destination, a *RREP* will be generated by the destination node as a response to *RREQ*. The *RREP* will be transmitted back to the originator of *RREQ* in order to inform the route. If an intermediate node has an active route towards destination, it can reply the *RREQ* with a *RREP*, which is called Gratuitous Route Reply. The intermediate node will also send an *RREP* to destination node. The *RREP* will be sent in reverse route of *RREQ* if a bidirectional link exists.

Route maintenance is performed with two additional messages: *Hello* and *RRER* messages. Each node broadcast *Hello* messages periodically to inform neighbours about its connectivity. The receiving of *Hello* message proves that there is an active route towards the originator. Each forwarding node should keep track of its continued connectivity to its active next hops. If a link to the next hop cannot be detected during a period of time-out, a *RRER* message will be broadcasted to inform the loss of connectivity. On receiving this *RRER*, usually a local repair will be performed just for maintenance. The expired route will be deleted after the confirmation of its unavailability.

From the security point of view, AODV requires at least two security attributes: sender authentication at each receiving node and routing message integrity. Message integrity is of the most concern in AODV routing. In route request broadcasting phase, each node has to check the originator sequence number in the *RREQ* packet with the one recorded in its routing table, and updates its routing table to the newest one; in route reply phase, instead of checking originator sequence number, each node check the destination sequence number and keeps it up-to-date. Any exploits of changing sequence number will result in routing loops.

Besides message alteration, spoofing is also a serious attack. A node forward RREP might claim itself to be someone else, misleading the receiving nodes falsely recording the fake identity as the next hop towards destination. This is another way of disrupting topology by creating route loops.

B. ARAN

ARAN was proposed by Sanzgiri et al in 2002, targeting to combat attacks including unauthorized participation, spoofed route signaling, alteration of routing messages, replay attacks, etc. Similar to other secure routing protocols, ARAN is also a security adds-on over on-demand routing protocols. It provides authentication, message integrity and non-repudiation as part of minimal security policy for ad hoc environment.

ARAN stands for Authenticated Routing for Ad hoc Networks. It is motivated to detect and protect against malicious actions by third parties and peers in an ad hoc environment. ARAN is a security scheme, which can be applied to any on-demand routing protocols. It takes the advantages of PKI based digital signature scheme to provide security features including authentication, message integrity and non-repudiation.

ARAN consists of three stages: a preliminary certification process, a mandatory end-to-end authentication stage and an optional stage providing secure shortest path. To deploy these three stages, ARAN requires the use of a trusted certificate server *T* and public key cryptography. Each node, before entering the network, must request a certificate from *T*, and will receive exactly one certificate after securely authenticating their identities to *T*.

Routing operations of ARAN are performed using three data structures: route discovery packet (*RDP*), reply packet (*REP*), and error packet (*ERR*). Each of them contains necessary routing information as well as the public key certificate. When a node wants to initiate a route discovery, it creates a signed *RDP* and broadcasts it to the next hop. The next hop node verifies the originator's signature. If it is authentic, it adds its own certificate and signs the whole packet again. The following hop node performs the same operation, however, after the verification of all the signatures of the received *RDP* it replaces previous hop node's signature with its own. Operations repeated until the packet reaches the target.

When the target node receives this *RDP*, it replies with a *REP*. This packet is in the same format of *RDP*, containing destination's signature and certificate. Each forwarding node verifies the signature, removes previous hop node's signature, and then adds its own outside the packet. If this route reply reaches the originator, it is guaranteed that the route found is authentic.

The authentication scheme provided by ARAN defends against exploits using modification, fabrication and impersonation. However, the use of public key cryptography is very costly. The computational overhead caused by signature generation and verification brings tremendous burden for mobile nodes. A group of malicious nodes may exploit this vulnerability to launch a deny-of-service attack by simply broadcasting large number of *RDP* packets. The receiving nodes have to exhaust their computational resources to verify the signature and then generate new ones. In addition, the extra bandwidth used to transmitting certificate is also another burden.

III. PRELIMINARIES

In this section, we introduce the one-time signature scheme to be used in the construction of our authentication scheme.

A. HORS

As we observed, since ARAN use public key cryptography to protect routing process, the time delay of signature generation and verification is significant. In general, significant time delay at each hop causes unacceptable route acquisition latency. Thus, we are looking for some digital signature schemes that maintain all the traits of conventional DSS, but are efficient enough in signature generation and verification.

The very first one-time signature scheme was introduced by Lamport in 1979 [7], to sign just 1 bit information. In 2002, Reyzin et al [10] proposed an one-time signature scheme, which is both efficient in signing and verification, and generating short signatures. This resulting scheme is called HORS, which stands for Hash to Obtain Random Subset. The major operation in signature generation is using a hashed message to obtain a random subset to form the signature.

HORS stands for Hash to Obtain Random Subset. It was proposed by Reyzin *et al* [10] in 2002, motivated to provide an efficient signing algorithm. HORS consists of three algorithms: key generation, signing and verification.

- **HORS Key Generation**

On constructing this scheme, several security parameters are predefined. To sign *b*-bit messages, we firstly pick *t* and *k* such that $\binom{t}{k} \geq 2^b$ and then choose a security parameter *l*, and a one-way hash function *f* that operates on *l*-bit strings. To generate public key, randomly generate *l*-bit string ($s_1, s_2, \ldots, s_t$). Let $v_i = f(s_i)$ for $1 \leq i \leq t$. The resulting public key is PK = ($k, v_1, v_2, \ldots, v_t$), private key is SK = ($k, s_1, s_2, \ldots, s_t$).

- **HORS Signature Generation**

To sign a message *m*, with secret key SK = ($k, s_1, s_2, \ldots, s_t$), firstly let $h = hash(m)$; then split *h* into *k* substrings $h_1, h_2, \ldots, h_k$, of length $\log_2 t$ bits each; finally, interpret each h_j as an integer i_j for $1 \leq j \leq k$. The resulting signature is $\sigma = (s_{i1}, s_{i2}, \ldots, s_{ik})$.

- **HORS Signature Verification**

The verification is the same as the signature generation. Suppose the verifier has the message *m*, signature $\sigma = (s'_{i1}, s'_{i2}, \ldots, s'_{ik})$, and public key PK = ($k, v_1, v_2, \ldots, v_t$). Firstly, let $h = hash(m)$; then split *h* into *k* substrings $h_1, h_2, \ldots, h_k$, of length $\log_2 t$ bits each and interpret each h_j as an integer i_j for $1 \leq j \leq k$. If for each j, $1 \leq j \leq k$, $f(s'_j) == v_{ij}$, accept the signature; otherwise, reject the signature.

In HORS, the public key component can be used multiple times. Signature generation requires only one call to hash function. Verification requires *k* calls to hash function. One impressive advantage of HORS is the shorter signature size. For their most efficient construction, the signature size can be reduced to 20480 bits.

B. One-Time Key Generation for Routing

Here, we describe the HORS one-time key generation process.

- **Notations:**

$h(), h, h^i()$ – one way function

$Sign_{Kn}$ – conventional digital signature generated by node *n*

$<>K_n^{-1}$ – one-time signature generated by node *n*

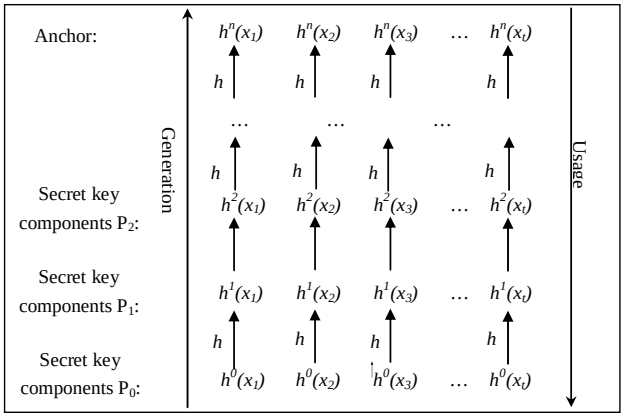

Figure 1. Secret key components hash chain.

- **Key Chain generation:**

Suppose that the decision has been made regarding security parameters *l*, *k* and *t* according to message length *b*.

1. Each node chooses *t* secret key components x_j (j=1,…,t) at random.
2. Each node creates a *n* hash chain of length *t* (see **Figure 1**):
3. Public key components are obtained through a one-way function *h*, namely $v_i = h(x_i)$. We assume that *h* is a hash function for simplicity.
4. Public key components are disclosed periodically.

Generating a set of one-time keys to sign routing messages has been discussed by Zhang in 1998 [15]. Two schemes called chained one-time signature scheme (COSP) and independent one-time signature scheme (IOSP) were proposed. These two schemes activate us to generate our novel scheme.

IV. AUTHENTICATED AODV ROUTING PROTOCOL

Based on the one-time signature scheme described above, we propose a security adds-on for AODV, which containing ARAN's authenticated routing features. This proposed protocol will provide following security properties:

1. The target node can authenticate the originator;
2. Each receiving node can authenticate its previous hop from which the routing message coming;
3. Each intermediate node can authenticate the sender for updating its routing table entry;
4. The hop count value is protected using hash chain. It cannot be reduced by a malicious node,

but could be increased more than one or retained unchanged, as in SAODV [14].

To achieve security features listed above, we firstly assume the existence of an offline CA, which issues certificate for each node when entering the network. Thus, each node possesses a public key and private key pair. The conventional digital signature will still be used to provide sender authentication, whereas the one-time signature will offer end-to-end authentication.

A. Public Key Handling

The public key in our proposed protocol is disseminated in two different ways. One aims at providing keys for authentication among neighbors. Another one tries to enable sender authentication during message transmission.

End-to-end authentication is achieved through neighbor authentication. Each node will generate a set of one-time keys as described in section 3.1. The one-time public key components are distributed locally among neighbors. Since one-time keys can only be used once or limited times, nodes need to update their one-time public keys periodically. To guarantee that each neighboring node has an authentic copy of node's public key, the very first public key, *anchor*, is distributed safely during system setup. When a node enters the network, it signs its anchor and broadcasts to its neighbors, along with its certificate. Thus, successive one-time public keys can be distributed in a more efficient manner by using *Hello* message, which is broadcasted periodically. The verification of updates is straightforward.

For example, the first secret key SK_1 is $(k, h^n(x_1), h^n(x_2), h^n(x_3), \ldots, h^n(x_t))$. The corresponding public key PK_1 is $(k, h^{n+1}(x_1), h^{n+1}(x_2), h^{n+1}(x_3), \ldots, h^{n+1}(x_t))$. The second secret key SK_2 is $(k, h^{n-1}(x_1), h^{n-1}(x_2), h^{n-1}(x_3), \ldots, h^{n-1}(x_t))$, thus the corresponding public key PK_2 is $(k, h^n(x_1), h^n(x_2), h^n(x_3), \ldots, h^n(x_t))$, which can be verified by hashing once and comparing to PK_1.

On the other hand, sender authentication is achieved through conventional digital signature. The sender's public key is contained in its certificate which is obtained when entering the network.

B. System Setup

This phase is used for initial key distribution (see **Figure 2**). Suppose when a mobile node enters the network, it is soon informed about the security parameters agreed in this network. It then chooses its secret key components and generates a hash chain according to section 3.1. Then it performs as follows:

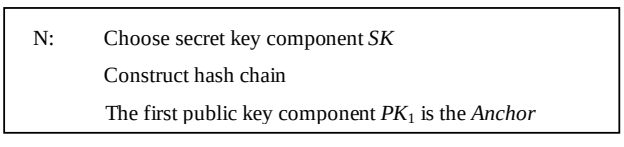

Figure 2. Initial key distribution and authentication (in System Setup)

C. Route Discovery

Route Discovery is performed as in **Figure 3**. When the originator (*S*) initiates a route discovery to a certain destination, it simply generates a signature over the RREQ, using conventional digital signature.

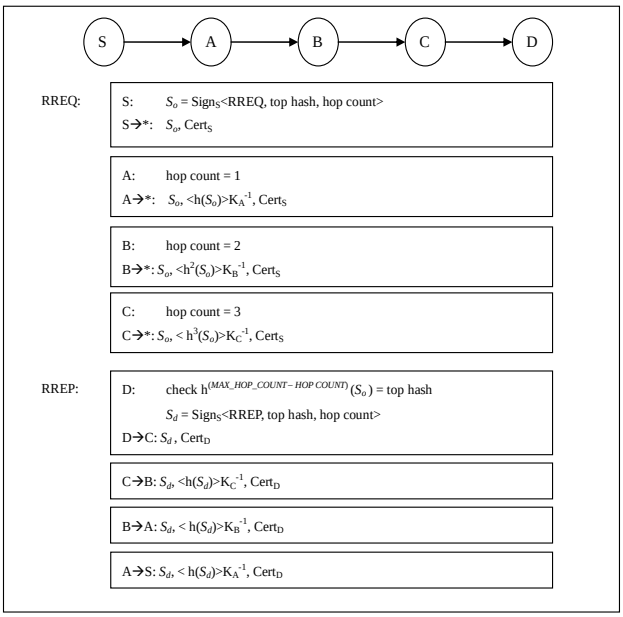

Figure 3. Route Request and Route Reply

Upon the first hop node (*A*) receives the RREQ, it firstly verifies the signature of the originator. If the signature is fine, the neighboring node hashes the received message S_o again and generates its own signature over it. This time, the signature is generated using HORS one-time signature scheme. Then the whole message is re-transmitted to second hop. From now on, there are two signatures. One is over S_o, another is over the hash of S_o.

Once the second hop node (*B*) receives this double signed RREQ, it firstly verifies the pervious hop (*A*) using public key of *A* (which might receive through *Hello* messages). If the one-time signature is fine, *B* hashes S_o one more time and creates a signature over the hash to replace the signature of *A*. Then this new message is broadcasted to next hop neighbors. Notice that the verification of conventional signature could be delayed. Only if both conventional signature and one-time signature are fine, does *B* update its routing table entry according to RREQ. These operations repeated until RREQ reaches the destination.

When RREQ reaches the destination, the destination node performs verifications the same as each intermediate node. Then a RREP is generated and signed the same as RREQ. Each intermediate node will transmit it back to the originator through the reverse route and same operations are performed along the route.

V. HANDLING GRATUITOUS ROUTE REPLY

In AODV, gratuitous route reply enables an intermediate node to reply RREQs which it has an active route towards the destination. This feature is optional in AODV, though turning on this feature will highly enhance the efficiency of routing discovery. However, to enable this feature, additional technique is needed. The

A→B: <A, $Token_A$>
$Token_A$ = <A, B, Pr_a, r_a, t_a>K_a^{-1}

Figure 4. Delegation Token

conceptual idea is that since we used digital signature to protect each routing message at each hop, for an intermediate node to reply RREQs instead of the destination, the intermediate node should be able to sign the RREQ properly on behalf of the destination.

A. Delegation Using Token

To solve this problem, we borrow the idea from proxy signature proposed by Varadharajan et al. [14], in which delegation is enabled by using a warrant. The warrant appears as a delegation token, containing the identities of primary signer and proxy signer, the privilege (Pr_a) given to proxy signer, an identifier (r_a) used by primary signer, and a timestamp (t_a). This delegation token is signed by the primary signer.

We simplify above delegation token into three fields (See **Figure 4**): the destination's identity, an identifier r_a and a timestamp t_a. It is possible because the token does not need to be designated to certain nodes. Any node that has received the token from a target is automatically proved to be having an active route towards the target. Otherwise, it would not be able to obtain this token. The token is signed by the creator using our IOS signature for our scheme.

The token enabled routing process is shown in **Figure 5**. If the gratuitous route reply option is turned on, nodes broadcasting RREQs must create tokens for gratuitous route reply delegation. The whole message including the token will be signed again, using the same public key as signing the token. Then, the originator broadcast the RREQ as usual.

Upon receiving the RREQ, node processes the authentication as normal. Then it checks the timestamp to see if the token has expired. If the token is valid, the nodes will store the token for future use.

The originator firstly checks if this RREP was created by destination or by intermediate node. If it is a gratuitous route reply, the originator checks the timestamp to determine if the route is still active. Then the token and the RREP will be authenticated as described before.

B. Delegation Using Transitive Signature Schemes References

In this section, we introduce another approach for enabling delegation by using transitive signature scheme.

Transitive signature scheme was firstly envisioned by Mciali and Rivest [6] in 2002. It was originally used to dynamically build an authenticated graph, edge by edge. The signer, having secret key *sk* and public key *pk*, can at any time pick a pair *i, j* of nodes and create a signature of {*i, j*}, thereby adding edge {*i, j*} to the graph. In addition, given a signature of an edge {*i, j*} and a signature of an

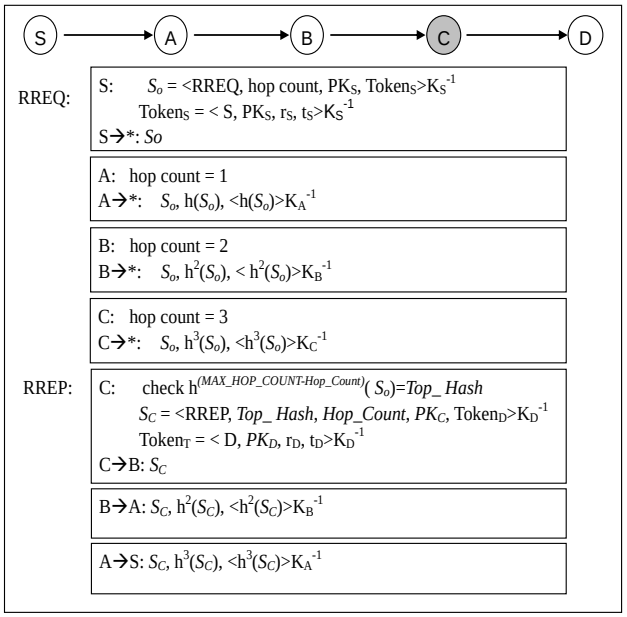

Figure 5. Token enabled route Request and Gratuitous Route Reply.

edge {*j, k*}, anyone in possession of the public key can create a signature of the edge {*i, k*}.

We make use of the transitive signature scheme proposed by Micali and Rivest [6] to construct our design.

- **Setup**

Each node in the network agrees with the following parameters:

- large prime *p* and *q* such that *q* divides *p-1*
- two generates *g* and *h* of subgroup G_q of order *q* $\in Z_p^*$ such that the base-*g* logarithm of *h* modular *p* is infeasible for others to compute.

Then each node n_i does the followed:

1. randomly choose two values x_i and y_i from Z_p^*;
2. compute $\alpha_i = x_i \bmod q$ and $\beta_i = y_i \bmod q$;
3. compute $v_i = g^{x_i} h^{y_i} \bmod p$;
4. broadcast α_i and β_i to node's neighbors.
5. upon the receipt of α_j and β_j from each neighbor, node *i* compute:
$$\alpha_{ij} = x_i - x_j \bmod q$$
and $$\beta_{ij} = y_i - y_j \bmod q$$
6. node *i* records in its memory the quadruple:
$$(v_i, v_j, \alpha_{ij}, \beta_{ij})$$

- **Sign**

To sign the path between node A and node B, node B must have received α_A, β_A, and v_A from node A. Then node B computes the signature as:

$$\alpha_{AB} = x_A - x_B \bmod q \quad \text{and}$$
$$\beta_{AB} = y_A - y_B \bmod q$$

Node B publishes the quadruple as the signature:

$$(v_A, v_B, \alpha_{AB}, \beta_{AB})$$

- **Verify**

Any node can verify the previous signature by checking:

$$v_A = v_B g^{\alpha_{AB}} h^{\beta_{AB}} \bmod q$$

- **Path Composing**

When the next hop node C receives the signature between node A and node B, it firstly verifies the validity of the signature in order to ensure that node B does have an active route towards node A. Then node C can generate a transitive signature over the received one so as to incorporate itself into the path.

Given signature $(v_A, v_B, \alpha_{AB}, \beta_{AB})$, node C retrieves the quadruple

$$(v_B, v_C, \alpha_{BC}, \beta_{BC})$$

and computes the new transitive signature

$$(v_A, v_C, \alpha_{AC}, \beta_{AC})$$

as:

$$\alpha_{AC} = \alpha_{AB} - \alpha_{BC} \bmod q = x_A - x_C \bmod q \quad \text{and}$$

$$\beta_{AC} = \beta_{AB} - \beta_{BC} \bmod q = y_A - y_C \bmod q$$

The signature for the path from node A to node C is:

$$(v_A, v_C, \alpha_{AC}, \beta_{AC})$$

The use of the transitive signature scheme to enable the route aggregation has one big benefit. It enables the authentication of both originator and gratuitous replier in one signature. In delegation by warrant, the token is signed with the routing packet by the gratuitous replier. Thus, the authentication of the gratuitous replier has to be done by verifying the conventional signature, and the token which is signed using conventional signature scheme has to be verified at the same cost. By using transitive signatures, the originator and replier can be authenticated at the same time.

However, the use of the transitive signature scheme to enable gratuitous reply authentication requires the cost of exchanging public key quadruples and computing the path signatures between neighboring nodes. It is considered to be the major drawback of this application.

VI. Discussion and Improvement

The most outstanding point of this scheme is the efficiency of one-time signature generation and verification at each hop. The same as HORS [10], each time, key generation requires t evaluation of one-way function. The secret key size is lt bits, and the public key size is $f_l t$ bits, where f_l is the length of the one-way function output on input of length l. The signature is kl bits long. There is a tradeoff between t and k, since the public key size will be linear in t, and the signature size and verification time will be linear in k.

The security of this scheme stems from the system setup phase. In this phase, a conventional digital signature is used to guarantee the authenticity of the first public key component. This can be achieved through using public key certificate issued by an offline CA, namely, each node must present a creditable identity when entering the network. The signature verification and generation may be inefficient, but since this message is broadcasted locally, it should be practical for each node.

The update of public key component is done along with *Hello* message, which is broadcasted periodically. Since the public key component comes from a hash chain, the verification is straightforward – the previous public key component is used to authenticate the new one. The trustworthiness of the new public component depends totally on the security of one-way hash function and the digital signature over *anchor*. The anchor is used only once. It is replaced by newly coming public key component after the first *Hello* message is broadcasted. In this way, nodes only need to do one hash to authenticate new public key component each time, which is much more efficient than hashing repeatedly back to anchor.

Sender authentication is performed with some compromise of efficiency, using conventional DSS. This method is much more secure than in SAODV, because in SAODV, the originator simply signs on its own public key without the support of PKI. Attackers can easily forge RREQ and RREP packets during transmission. On the other hand, the efficiency can be enhanced to some degree through the way that each node verifies conventional digital signature after broadcasting routing packets. Therefore, these will be no verification delay. Only both conventional signature and one-time signature is fine, will the routing table entry be updated.

Double signing over the received message does not provide more security than single signature from cryptographic point of view. Nevertheless, it provides non-repudiation hop-by-hop, which can be sued as an evidence for future intrusion detection. This thought comes from ARAN. It is considered as impractical because the use of conventional signature schemes. If there is a technique to produce even shorter signature in more efficient manner, this scheme can be extended to allow each node to sign on the received messages.

One significant drawback of one-time signature is that it can sign only predefined number of messages, which, in our scheme, is limited by the size of hash chain n. We generally consider it is not a serious problem, because nodes in MANET are mobile devices which are leaving and entering the network frequently. Consequently, the hash chain will be refreshed. In this sense, we can set n to a proper value according to network scale and average active time of nodes.

VII. Conclusion

This paper presented a novel scheme to implement ARAN protocol based on AODV routing protocol. However, it is more efficient than original ARAN in signature generation and verification by using HORS one-time signature in place of conventional digital signatures. We enable the protections for gratuitous route reply feature, under the concept of proxy signature's

delegation by warrant, as well as the route aggregation using transitive signature schemes. The warrant here is represented as a token, which contains creator's identity and public key, and is signed by the creator.

The security of our scheme needs to be enforced by performing conventional digital signature. With the help of asymmetric cryptography or public key certificate, we can ensure the authenticity of mobile nodes and the secure distribution of initial keys. Hence, the security of sub-sequential keys can be guaranteed by one way hash chain.

REFERENCES

[1] A. Burg. Ad Hoc Network Specific Attacks (pdf). In *Seminar. Ad hoc networking: concepts, applications, and security*, Technische Universität München, Nov. 2003.

[2] Y. C. Hu, D. Johnson, and A. Perrig. SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks (pdf). In *4th IEEE Workshop on Mobile Computing Systems and Applications (WMCSA' 02), June 2002*, pages 3-13, June 2002.

[3] Y. C. Hu, A. Perrig, and D. Johnson. Ariadne: A Secure On-demand Routing Protocol for Ad Hoc Networks (pdf). In *Proc. ACM MOBICOM*, Sep, 2002.

[4] Y. C. Hu, A. Perrig and D. B. Johnson. Rushing Attacks and Defense in Wireless Ad Hoc Network Routing Protocols (pdf). In *Proc. the 2003 ACM workshop on Wireless*, Sep. 2003.

[5] D. B. Johnson, D. A. Maltz and Y. C. Hu. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR). IETF INTERNET DRAFT, MANET working group, July. 2004. draft-ietf-manet-dsr-10.txt.

[6] S. Micali and R. Rivest. Transitive Signature Schemes. In B. Prneel, editor, *Topics in Cryptology – CT-RSA 2002*, volume 2271 of *Lecture Notes in Computer Science*, pages 236-243 Springer-Verlag, 2002.

[7] L. Lamport. Constructing digital signature from a one way function. Technical Report CSL-98, SRI International, October 1979.

[8] C. E. Perkins, E. M. Royer, and S. R. Das. Ad Hoc On-Demand Distance Vector (AODV) Routing. IETF INTERNET DRAFT, MANET working group. Feb. 2003. Draft-ietf-manet-aodv-13.txt.

[9] A. Perrig. The BiBa one-time signature and broadcast authentication protocol. In *8th ACM Conference on Computer and Communication Security, page 28-37. ACM, November 508, 2001*.

[10] L. Reyzin and N. Reyzin. Better Than BIBA: Short One-Time Signatures With Fast Signing and Verifying. In *Proc. 7th Australasian Conference on Information Security and Privacy, LNCS 2384*, Apr. 2002.

[11] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, and E. M. Royer. A Secure Routing Protocol for Ad Hoc Networks (pdf). Technical Report: UM-CS-2002-032, 2002.

[12] S. Xu, Y. Mu, and W. Susilo. Secure AODV Routing Protocol Using One-Time Signature. In *Proc. 1st International Conference on Mobile Ad-hoc and Sensor Networks (MSN 2005)*. Springer, LNCS 3794. Dec. 2005.

[13] V. Varadharajan, P. Allen, and S. Black. An Analysis of the Proxy Problem in Distributed Systems (pdf). In *Proceedings of the IEEE Symposium on Security and Privacy, 1991*, pages 255-275, May 1991.

[14] M. G. Zapata. Secure Ad hoc On-Demand Distance Vector (SAODV) Routing. *IETF INTERNET DRAFT, MANET working group*, Nov. 2004. draft-guerrero-manet-saodv-02.txt.

[15] K. Zhang. Efficient Protocols for Signing Routing Messages (pdf). In *Symposium on Network and Distributed Systems Security (NDSS '98)*, 1998.

Shidi Xu was born Chengdu, China on November 1, 1978. She received B.S. (Bachelor of Engineering) from the University of Electronic Science and Technology of China in 2001 and M. S. (Master of Information Systems) from the University of Wollongong in 2004. She is currently a candidate of Master of Computer Science by research in the University of Wollongong.

Yi Mu received his PhD from the Australian National University in 1994. He was previously with the School of Computing and IT at the University of Western Sydney as a lecturer and the Department of Computing at Macquarie University as a senior lecturer. He has been with the University of Wollongong since 2003. His current research interests include network security, electronic commerce security, wireless security, access control, computer security, and cryptography. He also previously worked at quantum cryptography, quantum computers, atomic computations, and quantum optics. His interest in other fields includes Internet computing, client/server software and web technology.

Yi Mu has served in program committees of a number of international conferences and editorial boards of several international Journals. He is a senior member of the IEEE and a member of the IACR.

Willy Susilo received a Ph.D. in Computer Science from University of Wollongong, Australia. He is currently a associate professor at the School of Information Technology and Computer Science of the University of Wollongong. He is the coordinator of Network Security Research Laboratory at the University of Wollongong. His research interests include cryptography, information security, computer security and network security. His main contribution is in the area of digital signature schemes, in particular fail-stop signature schemes and short signature schemes. He has served as a program committee member in a number of international conferences. He was the general chair of ACISP 2003. He is a member of the IACR.