UC-Secure Source Routing Protocol

Tao Feng[1], Xian Guo[1, 3], Jianfeng Ma[2], Xinghua Li[2]

[1]*School of Computer and Communication, Lanzhou University of Technology, China.*

[2]*Ministry of Education Key Laboratory of Computer Networks and Information Security, Xidian University, China.*

[3]*School of Computer and Math, Gansu Lianhe University, China*

Abstract: *The multi-path routing scheme provides reliable guarantee for mobile ad hoc network. A new method is proposed that is using to analyze the security of multi-path routing protocol within the framework of Universally Composable (UC) security. Based on the topological model that there exist adversarial nodes, the concept of plausible route is extended and the definition of plausible-route set is presented. Plausible-route set is used in description of the multi-path routing for Ad hoc network, and a formal security definition based on UC-RP is given. A provably Security Multiple Node-Disjoint Paths source routing (SMNDP) is proposed and address secure fault issue of MNDP in the active adversary model. The new approach shows that the security of SMNDP can be reduced to the security of the message authentication code and the digital signature. SMNDP implements the correctness of route discovery process, the authentication of nodes identifier and the integrality of route information.*

Keywords: *MANET, plausible route, UC-secure, SMNDP.*

1. Introduction

Routing is one of the most basic networking functions in mobile ad hoc networks (MANET). These networks are decentralized, which nodes act both as hosts and routers, forwarding packets for nodes that are not in transmission range of each other. Further, movement of nodes makes topology of the network to be dynamic. As a result, routing is one of the important issues in MANET. Routing in MANET can be accomplished through either single path or multiple paths. There are different types of the single-path routing protocols. One can distinguish reactive (e.g., DSR [1] and AODV [2]) and proactive (e.g., OLSR [3]). Protocols of the former category are also called on-demand protocols. Another type of classification distinguishes routing table-based protocols (e.g., AODV) and source routing protocols (e.g., DSR). Most of existing multi-path routing protocols are based on the single-path routing protocol. The single-path routing protocol is more simple, manageable and configurable than the multi-path routing protocol. In comparison with the single path routing, multi-path routing protocol has advantages in fault-tolerance, route-reliability, load-balancing and Quality-of-Service (QoS). Recently, multi-path routing attracts attention widely [4-6].

On the basis of disjointness, multiple paths can be the following categories: (1) node-disjoint; (2) link-disjoint; (3) neither node-disjoint nor link-disjoint. In this paper, we are mainly concerned with node-disjoint paths because using them one can best address the issues of fault-tolerance as well as load-sharing.

A new framework to identify node-disjoint paths is proposed in [7]. Based on the framework, Liu et al. proposed a routing protocol called Multiple Node-Disjoint Paths (MNDP) routing. In [8], Ash et al. presented a protocol that called Multiple Attempt Multi-path Routing (MAMR). MAMR is based on MNDP. In MAMR, the authors try to discover as many paths as possible in the first route discovery. Subsequent route discoveries identify paths in an incremental fashion using the approach of MNDP. However, MNDP and MAMR focus mainly on efficiency issues rather than security issues, we has found that they can’t defend against *active-n-m* attack [9].

Foundation item: Supported by the National Natural Science General Foundation of China under Grant No. 60573036, 60702059, Chinese National Programs for High Technology Research and Development (863), No. 2007AA01Z429, the Natural Science Grand Foundation of Gansu, China, No. 2007GS04823, the Ph.D. Programs Foundation of Lanzhou University of Technology No. BS14200901.

Biography: Tao Feng (1970-), male, Ph.D., Professor, research direction: information security. E-mail: fengt@lut.cn. Xian Guo (1971-), male, Ph.D. candidate, lectuer. His major research area is mobile ad hoc networks security. Jianfeng Ma (1963-), male, professor, Ph.D. supervisor. His major research interests include computer security, cryptography, mobile and wireless networks security. Xinghua Li (1978-), male, Ph.D., assistant professor, His major research interests include information security, trusted computing.

It is necessary to advocate a systematic approach for analyzing the security of ad hoc routing protocols, which is based on a rigorous mathematical model, in which precise definitions of security can be given and sound-proof techniques can be developed. The formal methods proposed in [10-11] have faults. Recently, a formal security framework for source routing of MANET is presented by Acs, Buttyan and Vajda [12-13], in which the concepts of reactive systems proposed by Pfitzmann and Waidner [14] are adopted, we call this framework ABV security model. In ABV, the description for the secure requirement of routing protocol is similar to the security definition of reactive systems. The definition of plausible route is proposed in network model where there exists adversarial nodes, and based on this definition and the computationally complex theory, the formal security definition for Ad hoc networks of routing protocol is presented. However, the security of single-path routing protocol is only concerned in ABV, and the composable security of routing protocols can't be described in asynchronous and concurrent networks environment. For example, multiple instances of routing protocol concurrently run in unpredictable environment of Internet.

Based on the computation model of interactive Turing machine, Canetti [15] proposed a framework of security defining for Universally Composable (UC) cryptographic protocols. To discuss the security of routing protocol, we advocate a security framework and call it UC-RP, in which UC security framework is adopted for MANET applications, and the security of the concurrent and composable routing protocols can be defined precisely. The security requirement of routing protocol is described by ideal functionality, the simulation paradigm is adopted when defining the security of routing protocol, and the universally composable security property can be realized by the composable operation theory in UC-RP. The routing protocol satisfied the security definition in UC-RP is called UC- secure routing protocol.

In this paper, based on the network-topology model where there exist the adversarial nodes, the concept of plausible route is extended. We proposed the concept of plausible-route set and the formal security definition of multiple node-disjoint paths routing protocol. And then, a UC-Secure Multiple Node-disjoint Paths (SMNDP) source routing protocol based on MNDP is proposed.

The main difference between MNDP and SMNDP is as follows: (1) In SMNDP, the error-check scheme is introduced in the transmission strategy of route quest for computing a new auxiliary path. (2) The cryptographic mechanisms, such as the digital signature and Message Authentication Code (MAC), are adopted in route reply message of SMNDP. The security of SMNDP can be reduced to the security of message authentication and signature schemes.

The work is the full version of [15].

2. Definition of Security for Routing

2.1. The Adversary Model

By controlling the adversarial node, the adversary can prevent the routing protocol from establishing multiple node-disjoint paths. Regarding the capabilities of the adversary, the followings is assumed:

1. Nodes are identified by identifiers in the neighbor discovery protocol and in the routing protocol. The identifiers are authenticated during neighbor discovery and therefore, the possibility of a Sybil attack [16] is excluded.
2. Wormholes [17] are detected at the neighbor discovery level, which means that nodes that are not within each other's radio range are not able to run the neighbor discovery protocol successfully.
3. The source and the destination of a route discovery process are not corrupted, and the adversary cannot modify or control all communications of the honest participants.
4. The adversary launches its attacks from a few adversarial nodes and can use all cryptographic keys that are necessary to authenticate those identifiers of adversarial nodes.
5. When the adversarial nodes are neighboring, the adversary is present at any compromised identifier.

2.2. The Network Model

We assume that the topology of MANET has been established, and the security bootstrap [19] of the network has been implemented before we discuss the security of routing protocol. MANET (in a given instance of time) is denoted by an undirected graph *G(V, E),* where *V* is the set of vertices and *E* is the set of edges. Each vertex represents either a single non-adversarial node or a set of adversarial nodes that can share information among themselves by communicating via direct wireless links or via out-of-band channels (these nodes are merged as a single adversarial node.). The former is called a non-adversarial vertex, while the latter is called an adversarial vertex. The set of adversarial vertices is denoted by V^*, and $V^* \subset V$. There is an edge between two non-adversarial vertices if the corresponding non-adversarial nodes established a wireless link between themselves by successfully running the neighbor discovery protocol. Furthermore, there is an edge between a non-adversarial vertex u and an adversarial vertex v^* if the non-adversarial node that corresponds to u established a wireless link with at least one of the adversarial nodes that correspond to v^*. Finally, there is no edge between two adversarial vertices in G. The rationale is that edges represent direct wireless links, and if two adversarial vertices u^* and v^* were connected, then there would be at least two adversarial nodes, one corresponding to u^* and the other corresponding to v^*, that could communicate with each other directly. That would mean that the adversarial nodes in u^* and v^* could share information via those two connected nodes, and thus, they should belong to a single vertex in G.

The set of all identifiers is denoted by L and the set of the compromised identifiers is denoted by L^*. The following is a labeling function D: $V \rightarrow 2^L$ (2^L is a family of subset of set L) that assigns a set of identifiers to each vertex in V:

$$\forall\, v \in V,\ D(v)=\begin{cases} l & v \in V \setminus V^* \\ L^* & v \in V^* \end{cases},\ \text{where } l \in L \backslash L^*$$

2.3. Configuration and Plausible-Route Set

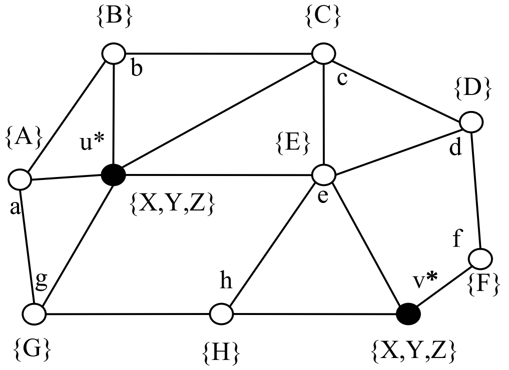

Fig.1 A configuration of network

A configuration (*conf*) of MANET based on the above network model is a triplet $(G(V, E), V^*, D)$. Fig.1 illustrates a configuration, in which the solid black dots are the vertices in V^* and each vertex is labeled with the set of identifiers that D assigns to it. Note that the vertices in V^* are not neighboring. It is assumed that a *conf* is static. Thus, we view the routing protocol as a distributed algorithm that operates on this static configuration. In fact, the securely minimum requirement that we require from the route discovery part of a multi-path secure source routing protocol is that it returns a set of multiple vertex-disjoint paths on this configuration.

In Fig.1. $\{A, B, C, D\}$ is an existence route between the vertices a and d. If there is no adversary, then a sequence $l_1, l_2, ..., l_n$ of identifiers is an existence route such that:

(1) each of identifiers $l_1, l_2, ..., l_n$ is different,

(2) there exists a sequence $v_1, v_2, ..., v_n$ of vertices in V such that $(v_i, v_{i+1}) \in E$ for all $1 \le i < n$ and $D(v_i)=l_i$ for all $1 \le i \le n$.

In Fig.1. $\{A,X,E,D\}$, $\{A,X,Y,E,D\}$ and $\{A,X,Y,Z,E,D\}$ are the same path route $\{a, u^*, e, d\}$ between the vertices a and d. The adversary can use all compromised identifiers in V^*, so a route indicated by a sequence of identifiers is not consistent with a route indicated by a sequence of vertices. To address the issue, a definition of plausible route is proposed in ABV. In this paper, the definition of plausible route is extended and the concept of plausible-route set is given.

Definition 1 (plausible route): Let *conf* $=(G(V, E), V^*, D)$ be a configuration, a sequence $l_1, l_2, ..., l_n$ of identifiers is a plausible route with respect to *conf* if there exists a sequence $v_1, v_2, ..., v_k$ $(2 \le k \le n)$ of vertices in V and a sequence $j_1, j_2, ..., j_k$ of positive integers such that:

(1) $j_1+j_2+...+j_k=n$,

(2) $\{l_{J_i+1}, l_{J_i+2}, ..., l_{J_i+j_i}\} \subseteq D(V_i)$ $(1 \le i \le k)$, where $J_i=j_1+j_2+...+j_{i-1}$ if $i>1$ and $J_i=0$, if $i=1$,and

(3) $(v_i, v_{i+1}) \in E$ $(1 \le i \le k)$.

As an example, let us consider again the configuration in Fig.1. It is easy to verify that $\{l_1, l_2, l_3, l_4, l_5, l_6\}=\{A，X，Y，Z，E，D\}$ is a plausible route, because it can be partitioned into four partitions $\{A\}$, $\{X, Y, Z\}$, $\{E\}$ and $\{D\}$, such that $\{A\} \subseteq D(a)$, $\{X, Y, Z\} \subseteq D(u^*)$, $\{E\} \subseteq D(e)$, $\{D\} \subseteq D(d)$ and vertices a, u^*, e and d form a simple path on the *conf*. In this example, $k= 4$, $j_1=1$, $j_2=3$, $j_3=1$, and $j_4=1$; furthermore, $J_1= 0$, $J_2= j_1=1$, $J_3= j_1 + j_2=4$, *and* $J_4= j_1 + j_2 + j_3=5$.

Definition 2 (*plausible-route set*)*:* Let *conf*$=(G(V, E), V^*, D)$ be a configuration, P is a plausible-route set between a given pair of vertices u and v, if P:

(1) each $p \in P$, p is a plausible route,

(2) any p_i, $p_j \in P$ $(i \neq j)$, p_i and p_j respectively correspond to sets of vertices sequence V_i and V_j, and $(V_i \cap V_j) \backslash \{u, v\} = \Phi$.

It is also easy to verify that $\{\{A, G, H, X, Y, F, D\}, \{A, X, Y, Z, E, D\}, \{A, B, C, D\}\}$ is a plausible-route set between the vertices a and d, because $\{A, G, H, X, Y, F, D\}$, $\{A, X, Y, Z, E, D\}$, and $\{A, B, C, D\}$ are plausible routes in this set and this set corresponds to a set of simple paths $\{\{a, g, h, v^*, f, d\}, \{a, u^*, e, d\}, \{a, b, c, d\}\}$.

To facilitate description, we call the two routes such as non-plausible route and non-auxiliary path error routes.

2.4. UC-RP Security Framework for Routing Protocol

It is a primary task of designing a secure routing protocol that the protocol can identify and establish an existence path route. A rigorous mathematic model represented a routing protocol is necessary, in which the formal definition of the secure requirement for routing protocol can be described. Based on the computationally complex theory, Goldwasser and Levin proposed a definition of security for cryptographic protocols. This definition is based on the indistinguishability of two families of binary random variable that one family of them represents a cryptographic protocol running in real-world model and another family of them represents cryptographic protocol in ideal-world model. The realization of “computational indistinguishability” is often called “cryptographic protocol simulation”.

Based on the extension of the concepts of the computational indistinguishability, the security definition of routing protocol is proposed in ABV. Based on the concept of reactive systems, the running process of routing protocol and the security requirement of routing protocol is described. The mathematic model of reactive systems comes from the I/O automata model, but no scheme can represent the capability of the adversary and the randomicity of the running process for protocol in the I/O automata model.

A new paradigm for defining the security of cryptographic protocols called Universally Composable security is proposed by Canetti. The salient properties of universally composable definitions of security are that they provide security guarantees even when a secure protocol are composed with an arbitrary set of protocols, or more generally when the protocol is used as a component of an arbitrary system. This is an essential property for maintaining security of cryptographic protocols in complicated and unpredictable communication network environment such as the Internet. In particular, universally composable definitions guarantee security even when an unbounded number of protocol instances are executed concurrently in an adversarially controlled manner. They guarantee non-malleability with respect to arbitrary protocols and more.

In UC, the model of computation based on “interactive Turing machines (ITMs)” denotes a system of cryptographic protocol. The security requirement of a protocol is expressed by “an ideal functionality”. The security definition of a protocol is described by “a simulation paradigm” and the universally composable security is implemented by “the composable operation theory” of protocols. A protocol that satisfies the security definition in UC is called a UC-secure protocol. Any n-party protocol π can be described by an interactive Turing machine system that consists of n interactive Turing machines p_i $(0 < i < n)$, in which an ITM stands for an honest participant of a protocol and the adversary entity is also modeled into an interactive Turing machine. In addition, an environment Z that a special ITM represents an external environment of a running protocol is defined in UC. Introduction of an environment Z can better describe this general case that protocols run asynchronously and concurrently. Z can arbitrarily interact with the adversary and the participants and Z’s view acquired in running process of this protocol is an output of the protocol.

In this paper, UC security model is introduced to discuss the security of routing protocol, and we propose a UC security framework for routing protocol, and call it UC-RP security framework. In UC-RP, we define the two computational models based on the well-known concept of interactive Turing machines: a real-world model and an ideal-world model. The real-world model describes the operation of the protocol with all its details in a particular computational model, and the ideal-world model describes an ideal protocol that realizes the ideal functionality.

Both of the models contain adversaries. The protocol is said to be secure if the real-world and the ideal-world model are equivalent, where the equivalence is defined as some form of indistinguishability (e.g. computational or statistical) from the point of view of the environmental machines Z. Namely, the effect of any real-world adversary in the real-world model can be simulated by an ideal-world adversary in the ideal-world model. Thus, the routing protocol running in real-world model is secure in UC-RP. Below, the provable-security theory of routing protocol is described in details:

1. *Protocol specification*, a protocol is a set of interactive Turing machines, and both the adversary and the environment Z are interactive Turing machines.
2. *Defining the adversary model.*
3. *The computational model represents the possible executions of a real protocol.* If a protocol π is a n-party, the protocol consists of a set of n interactive Turing machines and two entities such as the environment Z and the adversary A.
4. *The model of running protocol under the ideal state*. The ideal process consists of the ideal functionality F, the environment Z and the adversary S. In addition, there are n dummy parties that they send their inputs to the ideal functionality F and accept data coming from the ideal functionality F, if a protocol is an n-party protocol.
5. *Defining of the security*, we say that a protocol π securely realizes an ideal functionality F if the protocol π can emulate the ideal process that realize the ideal functionality F. That is to say, the routing protocol is secure if the two models are indistinguishable from the view of the environment Z.

2.4.1 The Real-World Model

The real-world model is denoted by $REAL_{conf,\pi, A,Z}(k, z, r)$, where $conf=(G(V, E), V^* ,D)$, π is a routing protocol that run in real-world model, A is a real-world adversary, k is security parameter, z is an input of the environment Z and $r=\{r_Z, r_A, r_1, r_2,\ldots, r_n\}$ is a set of random input for the system. $REAL_{conf,\pi,A,Z}(k, z, r)$ consists of a set $\{M_1, M_2, ..., M_n, A_1, ..., A_m, Z\}$ of interacting Turing machines, where the interaction is realized via common tapes. Each M_i represents a non-adversarial vertex in $V\backslash V^*$, and each A_j represents an adversarial vertex in V^*. All machines are probabilistic and they are activated by a hypothetic scheduler in rounds. The detailed behaviors of all machines are determined by the routing protocol under investigation.

The output of routing protocol is denoted by $OUT_{Real,conf,\pi,A,Z}(k, z, r)$, where r is initial parameters. In addition, let $X =\{OUT_{Real,conf,\pi,A,Z}(k, z, r)\}_{k\in N,Z\in \{0,1\}^*}$ denote the routing information ensemble when r is chosen uniformly at random. Let $OUT_{Real,conf,\pi,A,Z}$ denote $\{OUT_{Real,conf,\pi, A,Z}(k, z, r)\}_{k\in N,Z\in \{0,1\}^*}$.

2.4.2 The Ideal-World Model

The ideal-world model is denoted by $IDEAL_{conf,F,S,Z}(k, z, r)$, where $conf=(G(V, E), V^* ,D)$, S is a ideal-world adversary, k is security parameter, z is an input of the environment Z and $r=\{r_Z, r_A, r_1, r_2,...,r_n\}$ is a set of random input of the system. $IDEAL_{conf,F,S,Z}(k, z, r)$ also consists of a set $\{M'_1, M'_2, ..., M'_n, S, Z \}$ of interacting Turing machines and these machines work in the same way as they do in the real-world model. It is the main difference that the construction of the ideal functionality F in the ideal-world model can capture the requirement of routing protocol. In this paper, the minimum that we require from the routing protocol is that it returns only non-error route (plausible route and auxiliary path). In the ideal-world model, the ideal functionality F can easily identify and mark those route reply messages that contain error route. A marked route reply is processed by each machine M'_i in the same way as a non-marked one (i.e. the machines ignore the marker) except for the machine that initiated the route discovery process to which the marked route reply belongs. The initiator first performs all the verifications on the route reply that the routing protocol requires and if the message passes all these verifications, then it also checks if the message is marked as error route. If so, then it drops the message; otherwise, it continues processing (e.g., returns the received route). This definition of the ideal functionality F ensures that, in the ideal-world model, every route reply that contains an error route is caught and filtered out by the initiator of the route discovery. That is to say, the requirement that the routing protocol only returns non-error route is captured. We will denote the output of routing protocol by $OUT_{Ideal,conf,F,S,Z}(k, z, r)$. In addition, let $X=\{OUT_{Ideal,conf,F,S,Z}(k, z, r)\}_{k\in N,Z\in \{0,1\}^*}$ denote the routing information ensemble when r is chosen uniformly at random. Let $OUT_{Ideal,conf,F,S,Z}$ denote $\{OUT_{Ideal,conf,F,S,Z}(k, z, r)\}_{k\in N,Z\in \{0,1\}^*}$.

2.5. Definition of Security for Routing Protocol

The main steps of multi-path routing that identify the maximal set of node-disjoint paths in multiple route discoveries in an incremental fashion are as follows: in the first route discovery, the protocol identifies a reference path (*rp*) using a single path routing. In the second route discovery, the intermediate nodes corporately compute an auxiliary path (*ap*) according to the given transmission strategy of route request RREQ and the first reference path. And then, the reference path and the auxiliary path at the source node *B* are inspected and reorganized to generate two node-disjoint paths. In subsequent route discovery, node-disjoint paths acquired in the last route discovery are used as the reference paths in next route discovery. After the source node *B* gets a new auxiliary path, the reference paths and this new auxiliary path are again inspected and reorganized to generate a new set of multiple node-disjoint paths. Before the route discovery can find an auxiliary path, it will repeatedly do as the above route discovery. Finally, the set of multiple paths is a maximal set of node-disjoint paths.

The message format of the participant is (*sndr, rcvr, (msg, erf)*), *sndr* is the identifier of the sender, *rcvr* is the identifier of the receiver and *msg* is the actual protocol message. $erf \in \{undef, true, false\}$ is the error-route flag, and M_{ini}, M_{tar} are the identifiers of the initiator and the destination of the route discovery respectively. The details of the ideal functionality $F_{routing}$ for the routing protocol is showed in Fig.3.

Ideal Functionality $F_{Routing}$ of Routing Protocol

1. Upon receiving a *startup* message <*RREQ,* l_B, l_E > from protocol initiator M_{ini}, generates a unique subsession identifier *ssid*, and saves a record *Startup* (*ssid*, *RREQ*，l_B，l_E), sends < *ssid, RREQ,* l_B, l_E, *rp, ap*> to the adversary *S*,
2. Upon receiving a message *(ssid, RREQ,* l_B, l_E, *rp, ap）* from the adversary *S*, if $rp=\Phi$ (this is a first route discovery), player M_i generates a route request message *msg* and sends this message *msg* to all neighbors of M_i; if $rp\neq\Phi$ (several paths have been found before this route discovery), verifies that *ap* is a correctly auxiliary path. If this verification successes, player M_i generates a route request message *msg* and sends this message *msg* to all neighbors of M_i according to the transmission strategy of route request.
3. Upon receiving a message *(ssid, RREQ,* l_B, l_E, *rp, ap）* from the adversary *S*, if $rp=\Phi$ (this is a first route discovery), player M_{tar} generates a route reply message *msg* and unicasts this message *msg* to a neighbor of M_i on *ap*; if $rp\neq\Phi$ (several paths have been found before this route discovery), player M_{tar} generates a route reply message *msg* and unicasts this message *msg* to a neighbor of M_i on *ap.*
4. Upon receiving a message *(ssid, RREP,* l_B, l_E, *ap,msg)* from the adversary *S*, the initiator M_{ini} verifies:

   a) $rp=\Phi$, verifies that *ap* is a plausible route. If this verification successes, accepts this *ap* and generates the reference paths *rp*;

   b) $rp\neq\Phi$, verifies that *ap* is a plausible route and it is a correctly auxiliary path. If this verification successes, accepts *ap*. *rp* and *ap* are inspected and reorganized to generate a new set of node-disjoint paths.

Fig.3 **Ideal Functionality of Routing Protocol $F_{Routing}$**

Before copying a message (*sndr, rcvr, msg*) on any tape of any participant, $F_{routing}$ attaches a error-route flag *erf* to *msg*. This is done as the following way:

- If *msg* is a route request, then $F_{routing}$ sets *erf* to *undef.*

- If *msg* is a route reply and all routes carried by *msg* are non-error routes with respect to the configuration, then $F_{routing}$ sets *erf* to true.

- Otherwise, $F_{routing}$ sets *erf* to false.

When machine M'_i reads (*sndr, rcvr,* (*msg, erf*)) from itself tape, it verifies:

(1) *sndr* is its neighbor and $rcvr \in \{D(M'_i),*\}$ (* meaning a broadcast message). If these verifications are successful, then it performs the operations required by $F_{routing}$.

(2) *msg* is a route reply that belongs to a route discovery that was initiated by M'_i , then M'_i also checks if *erf*=false. If so, then M'_i performs the operations required by $F_{routing}$ and drops *msg*; otherwise, it continues processing it.

(3) If *msg* is not a route reply or M'_i is not the initiator, then *erf* is ignored. The messages generated by M'_i have no error-route flag attached to them, and they are placed in output tap of M_i.

Definition 3: (the routing protocol securely realizes the ideal functionality $F_{routing}$) Let $F_{routing}$ is an ideal functionality of a routing protocol and π is an *n*-party routing protocol. We say that π securely realizes $F_{routing}$ if for any real-world adversary *A* there exists an ideal-world adversary *S* such that two ensembles $OUT_{Ideal,conf,F,S,Z}$ and $OUT_{Real,conf,\pi,A,Z}$ is computational indistinguishability from a view of any environment *Z*, we have:

$$OUT_{Ideal,conf,F,S,Z} \approx OUT_{Real,conf,\pi,A,Z} \quad (2.1)$$

A routing protocol that can securely realize the ideal functionality $F_{Routing}$ is called a secure routing protocol in UC-RP security framework. Namely, there exists an ideal-world adversary S (also be called simulator) for any real-world adversary A. The adversary S should simulate this route if a routing protocol running in the real-world model returns a route. That is to say, the effect of any real-world adversary in the real-world model can be simulated by an ideal-world adversary in the ideal-world model. Because of the existence of the ideal functionality $F_{Routing}$, no ideal-world adversary can achieve that a error route is accepted in the ideal-world model, it follows that no real-world adversary can exist that can achieve that a error route is accepted with non-negligible probability in the real-world model because, if such a real-world adversary existed, then no ideal-world adversary could simulate it. In other words, if a routing protocol is secure in UC-RP, then it can return an error route only with negligible probability in the real-world model. This negligible probability is related to the fact that the adversary can always forge the cryptographic primitives (e.g., generate a valid digital signature etc.) with a very small probability.

2.6. Proof Technique

In order to prove the security of SMNDP, we have to find the appropriate ideal-world adversary S for any configuration *conf* and any real-world adversary A such that Definition 3.1 is satisfied. Due to the constructions of ABV, a natural candidate is $A=S$. Suppose that the initial parameters k and r of the two models are identical, then the operation of $REAL_{conf,\pi,\ A,Z}(k,\ z,\ r)$ can easily be simulated by the operation of $IDEAL_{conf,F,S,Z}(k,\ z,\ r)$. The following two cases can illustrate this fact:

(1) Let us assume that no message is dropped due to its error-route flag being false in $IDEAL_{conf,F,S,Z}(k, z, r)$. In this case, $REAL_{conf,\pi,\ A,Z}(k,\ z,\ r)$ and $IDEAL_{conf,F,S,Z}(k,\ z,\ r)$ are essentially identical, meaning that, in each step, the state of the corresponding machines and the content of the corresponding tapes are the same (apart from the error-route flags attached to the messages in $IDEAL_{conf,F,S,Z}(k, z, r)$). Namely,

$$OUT_{Ideal,conf,F,\ S,Z} = OUT_{Real,conf,\pi,A,Z}$$

(2) Suppose that some route reply messages are dropped in $IDEAL_{conf,F,S,Z}(k,\ z,\ r)$ due to their error-route flags being set to false, then $REAL_{conf,\pi,\ A,Z}(k, z, r)$ and $IDEAL_{conf,F,S,Z}(k, z, r)$ may end up in different states and their further steps may not match each other, since those messages are not dropped in $REAL_{conf,\pi,\ A,Z}(k, z, r)$. We call this situation a simulation failure. In case of a simulation failure, it might be that $OUT_{Ideal,conf,F,\ S,Z} \neq OUT_{Real,conf,\pi,A,Z}$. Nevertheless, the definition of security can still be satisfied, if simulation failures occur only with negligible probability.

Hence, when trying to prove the security of routing protocol, we must try to prove that for any configuration *conf* and adversary A, the event of dropping a route reply in $IDEAL_{conf,F,S,Z}(k,\ z,\ r)$ due to its error-route flag being set to false can occur only with negligible probability.

3. MNDP and the Security of MNDP

3.1. Overview of MNDP

Identifying a maximal set of node-disjoint paths between a given source and destination is a challenging task in MANET. A graph theoretic framework to identify node-disjoint paths is proposed by Liu et al. in [7]. The main idea of this framework is that a undirected graph $G(V,\ E)$ denoted an ad hoc network is transformed to a corresponding unit capacity flow network $G^F(V^F,\ E^F)$. It is proved that the problem of finding the maximum number of node-disjoint paths in original network $G(V,\ E)$ is equivalent to the problem of finding the maximum flow in flow network $G^F(V^F,\ E^F)$. This equivalence provides guarantee to compute a maximal set of node-disjoint paths. Based on the framework, Liu proposed a routing protocol called Multiple Node-Disjoint Path (MNDP). Liu have used MNDP to discover two node-disjoint paths. However, it can potentially be extended to find all node-disjoint paths that exist between a given pair of nodes. Furthermore, the fact that MNDP is guaranteed to discover multiple node-disjoint paths has been proved in [7] using concepts of flow networks.

Suppose that the source node of the route discovery is B and the destination node of the route discovery is E. The main steps of MNDP are as follows: In the first route discovery, the protocol identifies a reference path using a single path routing such as Dynamic Source Routing (DSR) [1]. In the second route discovery, based on the Ford-Fulkerson approach [20] that computes the maximum flow in the flow network G^F, the intermediate nodes corporately compute an auxiliary path according to the transmission strategy of MNDP for RREQ and the first reference path. And then, the reference path and the auxiliary path at the source node *B* are inspected and reorganized to generate two node-disjoint paths. In subsequent route discovery, node-disjoint paths acquired in the last route discovery are used as the reference paths by MNDP. The source node *B* gets a new auxiliary path. Again, the reference paths and this new auxiliary path are inspected and reorganized to generate a new set of multiple node-disjoint paths. Before the route discovery can find an auxiliary path, it will repeatedly do as the above route discovery. Now, the set of multiple paths is a maximal set of node-disjoint paths according to Liu's conclusion.

The transmission strategy for the route request of the intermediate nodes is the crucial part of MNDP. We make assumption as follows: The source node of the route discovery is *B* and the destination node of the route discovery is *E*. If the node *p* (*p* may be the source node) is the sender of the route request RREQ and the node *t* (*t* may be the destination node) is the receiver of the route request RREQ, the node *t* perform the corresponding transmission strategy according to whether the node *t* is present on the reference path rp_i ($1 \leqslant i \leqslant n$, *n* is the number of paths in the reference paths *rp*) when the node *t* receives the route request RREQ from the node *p*. The details of the transmission strategy are shown in table 1.

Sender *p* , Receiver *t*		**The transmission strategy of *t***		**No.**
The relation of Nodes *p, t* and rp_x	**The location relation of nodes *p, t* on a reference path rp_i**	**The style of forwarding**	**Identifier**	
$p,t \in rp_i$	*p* is a successor of *t*	Broadcasts	Appends	1
	p is a predecessor of *t*	Discards	No	2
	P and t are not neighboring on rp_i	Unicasts to the predecessor of *t* on rp_i	Appends	3
$p \in rp_i$, $t \in rp_j$	no	Unicasts to the predecessor of *t* on rp_i	Appends	4
$p \in rp_i$, $t \notin rp_x$	no	Broadcasts	Appends	5
$p \notin rp_x$, $t \in rp_i$	no	Unicasts to the predecessor of *t* on rp_i	Appends	6
$p,t \notin rp_x$	no	Broadcasts	Appends	7

Table 1：The transmission scheme of route quest for computing the auxiliary path in MNDP

In table 1, rp_x ($1 \leqslant x \leqslant n$, *n* is the number of paths in the reference paths *rp*) is some reference path in the reference paths *rp*. The transmission strategy is described as follows:

(1) If the identifier of the node *t* is present on the reference path rp_i ($1 \leqslant i \leqslant n$), it checks whether the node *p* from which it received RREQ is its successor or predecessor on rp_i. The node *t* acts as follows.

(1.1) If the node *p* is neither a successor nor a predecessor of node *t* on rp_i ($1 \leqslant i \leqslant n$), it appends its own identifier on the auxiliary path *ap* and unicasts it to its predecessor on rp_i.

(1.2) If the node *p* is successor of the node *t* on the reference path rp_i ($1 \leqslant i \leqslant n$), it appends its own identifier on the auxiliary *ap* and broadcasts the RREQ to its neighbors.

(1.3) If the node *p* is predecessor of the node *t* on the reference path rp_i ($1 \leqslant i \leqslant n$), the node *i* then drops the RREQ.

(2) If identifier of the node *t* is not present on any reference path, it appends its own identifier on the auxiliary path and broadcasts the RREQ to its neighbors.

3.2. Analysis of the Security for MNDP

The attack against Ad hoc networks is classified into two main classes: passive and active [9]. The passive attacker only eavesdrops on the network. It mainly threats against the privacy or anonymity of communication, rather than against the functioning of the network or its routing protocol. An active attacker can inject packets into the network and generally also eavesdrop. In the attacker model *Active-n-m*, *n* represents the number of nodes the attacker has compromised, and *m* is the number of the nodes the attacker owned. MNDP cannot defend against *active-n-m* attack.

In Fig.2a, let us suppose that the adversary doesn't exist in the network and *rp*={{1, 4}, {2, 5}} is a set of reference paths for the third route discovery from the source *B* to the destination *E*. In the third route discovery, the node 5 appends its identifier in *ap* and unicast the RREQ to the node 2 according to the transmission strategy of MNDP (the rule 4 in table 1) when the node 5 receives the route request RREQ that contains *ap*={3, 4, 1} from the node 1. Similarly, the node 2 appends its identifiers in *ap* and broadcast the RREQ to its all neighbors (the rule 1 in table 5) when it receives the RREQ that contains *ap*={3, 4, 1, 5} from the node 5. As a result,

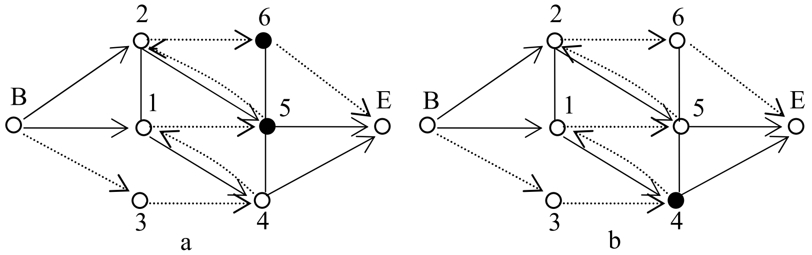

Fig.2 Scenario where attacks against MNDP. Adversarial nodes are represented by solid black dots. The labels assigned to nodes are identifiers that used to identify node. A route between a given pair of nodes is denoted by a sequence of identifiers.

a new auxiliary path *ap*={3, 4, 1, 5, 2, 6} is returned by the destination node *E*. Finally, the reorganization of *ap* and *rp* at the source node *B* generates a new set of node-disjoint paths {{1, 5}, {2, 6}, {3, 4}}.

Because not any cryptographic mechanism and error-check scheme are adopted in MNDP, the active adversary easily damages the route discovery process for MNDP of the auxiliary path via compromising the intermediate node. For example, the active adversary attacks against MNDP by violating the transmission strategy for MNDP of the route request or modifying the message of the reference path and the auxiliary path on adversarial node. As a result, the task of establishing multiple node-disjoint paths can't be accomplished. Namely, the reorganization of the reference path and the auxiliary path acquired in a route discovery can't generate a new set of multiple node-disjoint paths. In this paper, we view the path as a non-auxiliary path when the above case occurs. Below, we will illustrate the secure flaws of MNDP.

Attack scenario 1: Fig.2a, we assume that *rp*={{1, 4}, {2, 5}} is a set of reference paths found between a given source *B* and destination *E* in the first two route discovery. In the third route discovery, after the adversary appends the compromised identifier 5 in *ap*, it doesn't unicast the route request RREQ to the node 2 rather than to the node *E* such that *ap*={3, 4, 1, 5} when the adversary receives the route request RREQ that contains *ap*={3, 4, 1} from the node 1 on the compromised node 5. Clearly, the adversary violates the transmission rule 6 in table 1. And then, the destination node *E* generates a route reply RREP that contains *ap*={3, 4, 1, 5} and sends the RREP back to the source node *B* on the reverse of *ap*. Therefore, the reorganization of *rp* and *ap* found in this route discovery generates a set of routes {{1, 5}, {2, 5}, {3, 4}} when the source node *B* receives this RREP. Apparently, there exist node-intersecting routes {1, 5} and {2, 5} in this set, MNDP can't defend against this simple *active-1-1* attack.

Attack scenario 2: Fig.2a, we still assume that *rp*={{1, 4}, {2, 5}} is a set of reference paths found between the given source *B* and the destination *E* in the first two route discovery, and *ap*={3, 4, 1, 5, 2, 6} is an auxiliary path found in this route discovery. The adversary modifies *ap*={3, 4, 1, 5, 2, 6} into *ap*={3, 4, 1, 5} and unicasts the RREP that contains *ap*={3, 4, 1, 5} to the node 1, when the adversary receives the RREP that contains *ap*={3, 4, 1, 5, 2, 6} on the compromised node 5. And then, when the source node *B* receives this RREP, that is, *ap*={3, 4, 1, 5}, the reorganization of *rp* and *ap* found in this route discovery generates a set of routes {{1, 5}, {2, 5}, {3, 4}}. There also exist two node-intersecting routes in this route set. Therefore, MNDP can't defend against this *active-2-2* attack.

Attack scenario 3: Fig.2b, we again assume that *rp*={{1, 4}, {2, 5}} is a set of reference paths found between a given source *B* and destination *E* in the first two route discovery. When the adversary receives a route request RREQ that contains *rp*={{1, 4}, {2, 5}} and *ap*={3} on an adversarial node 4, *rp*={{1, 4}, {2, 5}} that contains in the RREQ is modified into *rp*={{1, 4}, {2, 6}} by the adversary on adversarial node 4, and then the adversary appends 4 in *ap* and unicasts this RREQ to the node 1. The node 1 and 5 respectively append 1 and 5 in *ap* and broadcast this request to their all neighbors when they receive this RREQ (the transmission rule 1 and 5 in table 1). And then, the destination node *E* generates a route reply RREP that contains *ap*={3, 4, 1, 5} and sends the RREP back to the source node *B* on the reverse of *ap*. A new route set {{3, 4}, {1, 5}, {2, 5}} is generated when the reference paths *rp*={{1, 4}, {2, 5}} and the auxiliary path *ap*={3, 4, 1, 5} are inspected and reorganized at the source node *B*. Clearly, this *active-1-2* attack against MNDP is easily and successfully implemented by the adversary.

4. Solution of SMNDP

The relation of nodes l_{m-1}, l_m, l_{m+1} and rp^*		**The relation of nodes l_{m-1}, l_m, l_{m+1} and rp_x**	**The relation of nodes l_{m-1}, l_m, l_{m+1} on some path rp_x**	**The transmission strategy of l_{m+1}**		
Receiver l_{m+1}	***The last two nodes l_{m-1}, l_m in ap***			**The style of forwarding**	**Identifier**	**No.**
$l_{m+1} \notin rp^*$	$l_{m-1} \notin rp^*, l_m \notin rp^*$	No	No	Broadcasts	Appends	1
	$l_{m-1} \in rp^*, l_m \notin rp^*$	$l_{m-1} \in rp_i$;	No	Broadcasts	Appends	2
	$l_{m-1} \notin rp^*, l_m \in rp^*$	$l_m \in rp_i$;	No	Drops (7)	No	3
	$l_{m-1} \in rp^*, l_m \in rp^*$	$l_{m-1} \in rp_i$; $l_m \in rp_j$	No	Drops (5)	No	4
		$l_{m-1}, l_m \in rp_i$	$l_{m-1}$is a successor of l_m	Broadcast	Appends	5
			l_{m-1} is a predecessor of l_m	Drops (3)	No	6
			l_{m-1} is not a neighbor of l_m	Drops (4)	No	7
$l_{m+1} \in rp^*$	$l_{m-1} \notin rp^*, l_m \notin rp^*$	$l_{m+1} \in rp_i$;	No	Unicasts to the predecessor of l_{m+1} on rp_i	Appends	8
	$l_{m-1} \in rp^*, l_m \notin rp^*$	$l_{m+1} \in rp_i$	Doesn't consider	Unicasts to the predecessor of l_{m+1} on rp_i	Appends	9
	$l_{m-1} \notin rp^*, l_m \in rp^*$	$l_m \in rp_i$; $l_{m+1} \in rp_j$	No	Drops (7)	No	10
		l_m, $l_{m+1} \in rp_i$	l_m is a successor of l_{m+1}	Broadcast	Appends	11
			l_m is a predecessor of l_{m+1}	Drops (7)	No	12
			l_m, l_{m+1} are not neighboring	Drops (7)	No	13
	$l_{m-1} \in rp^*, l_m \in rp^*$	$l_{m-1}, l_m \in rp_i$, $l_{m+1} \in rp_j$	l_{m-1} is a successor of l_m	Unicasts to the predecessor of l_{m+1} on rp_j	Appends	14
			l_{m-1} is a predecessor of l_m	Drops (3)	No	15
			l_{m-1}, l_m are not neighboring	Drops (4)	No	16
		$l_{m-1} \in rp_i$, $l_m, l_{m+1} \in rp_j$	l_m is a successor of l_{m+1}	Broadcast	Appends	17
			l_m is a predecessor of l_{m+1}	Drops (5)	No	18
			l_m, l_{m+1} are not neighboring	Drops (5)	No	19
		$l_{m-1}, l_{m+1} \in rp_i$, $l_m \in rp_j$	Doesn't consider	Drops (5)	No	20
		l_{m-1}, l_m , $l_{m+1} \in rp_i$	l_{m-1} is a successor of l_m l_m is a successor of l_{m+1}	Broadcast	Appends	21
			l_{m-1} is a predecessor of l_m or l_m is a predecessor of l_{m+1}	Drops (3)	No	22
			l_{m-1}, l_m are not neighboring or l_m, l_{m+1} are not neighboring	Drops (4)	No	23
		$l_{m-1} \in rp_i$, $l_m \in rp_j$, $l_{m+1} \in rp_k$	No	Drops (5)	No	24

Table 2. The transmission strategy of route request for computing the auxiliary path in SMNDP

No any cryptographic primitives and error-check scheme are used in MNDP, so an active attacker can successfully prevent the routing protocol from identifying the set of multiple node-disjoint paths by violating the transmission strategy of the route request for computing the first reference path and a new auxiliary path, or modifying the reference paths and the route list in the route request message and the route reply message for computing a new auxiliary path.

The main differences between MNDP and SMNDP: (1) based on the network model in section 2 and the fact that the adversarial vertices are non-neighboring on *conf*, the error-check scheme is introduced in the transmission strategy of route request for computing a new auxiliary path. We extend the transmission strategy of the route request for two nodes in MNDP into the transmission strategy of the route request for three nodes. The error-check scheme can check out the faulty behavior that the adversary violates the transmission strategy of the route request for SMNDP. (2) In the algorithm of the route reply for SMNDP, the two cryptographic mechanisms such as the digital signature and Message Authentication Code (MAC) are adopted. The digital signature provides guarantee to return a plausible route, and the attack that modifies the reference paths in route request message can be found by verifying MAC.

The protocol SMNDP mainly consists of three parts: the algorithm of route request for SMNDP, the algorithm of route reply for SMNDP and the algorithm of reorganization at the source node. The algorithm of reorganization is similar to that of MNDP, so we don't discuss it in this paper. Below, we will describe the first two parts in detail.

4.1 The Algorithm of Route Request for SMNDP

The algorithm of route request for SMNDP includes two parts such as the algorithm of route request for the first reference path and the algorithm of route request for computing the auxiliary path. It is similar to MNDP that SMNDP identifies a reference path using a singe path routing such as DSR [1] in the first route discover. You can see [7]. We mainly illustrate the transmission strategy of route request for SMNDP.

The strategy that the intermediate node forwards the route request RREQ for computing an auxiliary path is a crucial part of SMNDP. We assume that *B* is the source node of the route discovery and *E* is the destination node of the route discovery. Let us further assume that the intermediate node l_{m+1} has received a route request RREQ that contains the auxiliary path $ap=(l_1, ..., l_{m-1}, l_m)$. Likely, here l_m is a source node *B* and l_{m+1} is a destination node *E*. The node l_{m+1} performances the corresponding transmission strategy according to the location relationship of this three nodes such as l_{m-1}, l_m , l_{m+1} on the reference paths. The transmission strategy of RREQ is showed in Table 2. In table 2, rp^* is any path in the reference paths *rp* and rp_x denotes the path that indexed by x ($1 \leqslant x \leqslant n$, n is the number of the paths in the reference paths *rp*) in the reference paths *rp*. The no. *n* in the bracket indicates that the error-check scheme of SMNDP checks out the node l_m violates the rule *n* of MNDP.

4.2. The Algorithm of Route Reply for SMNDP

Let us suppose that the current reference paths are *rp*, the algorithm of route reply for SMNDP is shown in Fig.4.

4.3. The Algorithm of Route Discovery for SMNDP

The route discovery process of SMNDP is shown in Fig.5.

5. The Security Analysis of SMNDP

Lemma 1: Based on the network model that there exists adversary nodes, if the signature scheme used by SMNDP is secure, SMNDP returns a non-plausible route on any configuration *conf* only with negligible probability in ideal-world model of UC-RP.

Proof: Let us suppose that the route $ap=\{l_1, l_2,...,l_n\}$ returned by SMNDP is a non-plausible route on any configuration *conf*. The route reply message that contains the route *ap* is the following:

$$msg{=}{<}ssid,\ RREP,\ l_B,\ l_E,\ ap,\ MAC_{K_{B,E}}(ssid,\ rp),\ Sig_{l_E},\ Sig_{l_n},\ \ldots,\ Sig_{l_1}\ >$$

Further, let us suppose that *msg* passes all the verifications required by SMNDP at the source node *B*, which means that all signatures in *msg* is correct, the source node *B* has a neighbor that uses the identifier l_1. Recalling that, by definition of configuration *conf*, adversarial vertices cannot be neighbors. In addition, each non-adversarial

The Algorithm of Route Reply for SMNDP

a) ***The destination node E processes the route request RREQ*:** when the destination node E receives the route request RREQ

$<sid, RREQ, l_B, l_E, rp, ap=(l_1, ..., l_i, ..., l_n)>$

The destination node E generates a signature Sig_{l_E} on l_B, l_E, and ap and generates a message authentication code $MAC_{K_{B,E}}(sid, rp)$ on sid, rp with private key $K_{B,E}$ shared by the destination node E and the source node B. And then it generates a route reply message RREP

$< sid, RREP, l_B, l_E, ap, MAC_{K_{B,E}}(sid,rp), Sig_{l_E} >$

Finally, the destination node E unicasts the RREP to the last node l_n, and the destination E drops the other copies of the RREQ.

b) ***The intermediate node i processes the route reply RREP***: when the intermediate node i receives the route reply RREP, it verifies that its identifier l_i is in node list ap, and that the preceding identifier (or that of the initiator, if there is no preceding identifier in the node list ap) and the following identifier (or that of the target, if there is no following identifier in the node list ap) belongs to neighboring nodes of the node i. The node i also verifies that the digital signatures in the reply are valid and that they correspond to the following identifiers in the node list ap and to the target. If these verifications fail, then the route reply RREP is dropped. Otherwise, the intermediate node i generates a signature on l_B, l_E, ap and a new route reply message.

$<sid, RREP, l_B, l_E, ap, MAC_{K_{B,E}}(sid, rp), Sig_{l_E}, Sig_{l_n}, ..., Sig_{l_i} >$

c) ***The source node B processes a route reply message RREP***:

$< sid, RREP, l_B, l_E, ap, MAC_{K_{B,E}}(sid,rp), Sig_{l_E}, Sig_{l_n}, ..., Sig_{l_1} >$

(1) $rp=\Phi$, the source node B verifies that the first node in the node list is its neighbor, and verifies all of signatures in RREP. If these verifications success, the source node B accepts this ap as the first reference path; otherwise, it drops RREP and initiates a new route discovery.

(2) $rp\neq\Phi$, the source node B verifies that the first node in the node list is its neighbor, and verifies all of signatures in RREP. In addition, the source node B verifies the message authentication code $MAC_{K_{B,E}}(sid,rp)$ according to the current reference paths rp. If these verifications success, the source node B accepts ap as a new auxiliary path; otherwise, it drops the RREP and initiates a new route discovery.

Fig.4. The Algorithm of Route Reply for SMNDP

The Algorithm of Route Discovery for SMNDP

a) ***The source node B broadcasts a route request RREQ:***

$<sid, RREQ, l_B, l_E, rp, ap>$

Here, l_B, l_E are respectively identifiers of the source node B and the destination node E. sid is identifier of route discovery, rp is a set of multiple node-disjoint paths found before this route discovery in route cache. $ap=\Phi$, it will be used to record the first reference path and the auxiliary path in route discovery process.

b) ***The intermediate node i processes the route request RREQ:***

$<sid, RREQ, l_B, l_E, rp, ap=(l_1, ..., l_{m-1}, l_m)>$

The intermediate node i drops the RREQ if it has processed the RREQ. Otherwise, if $rp=\Phi$, the intermediate node i forwards the RREQ to the destination node E according to the algorithm of the route request for DSR. If $rp\neq\Phi$, the intermediate node i forwards the RREQ to the destination node E according to the transmission strategy of route request in table 2.

c) The algorithm that the destination node E generates a route reply RREP, and that the intermediate node i processes route reply RREP is shown in Fig.4.

d) The source node B processes the route reply RREP and performs the algorithm of reorganization: The algorithm that the source node B processes the RREP is shown in Fig.4. The algorithm of reorganization at the source node is similar to that of MNDP. You can see [7].

e) Repeat (a)-(d), until k paths that protocol need have been found or no new auxiliary path can be found.

Fig.5. The Algorithm of Route Discovery for SMNDP

vertex has a single and unique non-compromised identifier assigned to it. It follows that every route, including (l_B, l_1, ..., l_n, l_E), has a unique meaningful partitioning, which is the following: Each non-compromised identifier, as well as each sequence of consecutive compromised identifiers, should form a partition. Let $P_1, P_2, ...,P_k$ be the unique meaningful partitioning of the route （l_B, l_1, ..., l_n, l_E）.The fact that this route is non-plausible implies that at least one of the following two statements holds:

- Case 1. There exist two partitions $P_i=\{l_j\}$ and $P_{i+1}=\{l_{j+1}\}$ such that both l_j and l_{j+1} are non-compromised identifiers and the corresponding non-adversarial vertices are not neighbors.
- Case 2. There exist three partitions $P_i=\{l_j\}$, $P_{i+1}=\{l_{j+1}, ..., l_{j+q}\}$, and $P_{i+2}=\{l_{j+q+1}\}$ such that l_j and l_{j+q+1} are non-compromised and $l_{j+1}, ..., l_{j+q}$ are compromised identifiers, and the non-adversarial vertices that correspond to l_j and l_{j+q+1}, respectively, have no common adversarial neighbor.

We show that in both cases, the adversary must have forged the digital signature of a non-adversarial machine.

In Case 1, machine l_{j+1} does not sign the route reply, since it is non-adversarial and it detects that the identifier that precedes its own identifier in the route does not belong to a neighboring machine. Hence, the adversary must have forged $Sig_{l_{j+1}}$ in msg.

In Case 2, the situation is more complicated. Let us assume that the adversary has not forged the signature of any of the non-adversarial machines. Machine l_j must have received

$$msg'=<ssid, RREP, l_B, l_E, ap, MAC_{K_{B,E}}(ssid, rp), Sig_{l_E}, Sig_{l_n}, ..., Sig_{l_{j+1}}>$$

from an adversarial neighbor, say, A, since l_{j+1} is compromised and thus a non-adversarial machine would not send out a route reply message with $Sig_{l_{j+1}}$. In order to generate *msg'*, machine A must have received

$$msg''=<ssid, RREP, l_B, l_E, ap, MAC_{K_{B,E}}(ssid, rp), Sig_{l_E}, Sig_{l_n}, ..., Sig_{l_{j+q+1}}>$$

because, by assumption, the adversary has not forged the signature of l_{j+q+1}, which is non-compromised. Since A has no adversarial neighbor, it could have received *msg''* only from a non-adversarial machine. However, the only non-adversarial machine that would send out *msg''* is l_{j+q+1}. This would mean that A is a common adversarial neighbor of l_j and l_{j+q+1}, which contradicts the assumption of case 2. This means that our original assumption cannot be true, and hence, the adversary must have forged the signature of a non-adversarial machine.

However, the adversary can forge the signature of a non-adversarial machine only with negligible probability in both cases, if the digital signature used by SMNDP is secure. So, SMNDP returns a non-plausible route only with negligible probability in ideal-world model of UC-RP.

Lemma 2: Based on the network model that there exists adversary nodes, if Message Authentication Code (MAC) used by SMNDP is secure, SMNDP returns a non-auxiliary path on any configuration *conf* only with negligible probability in ideal-world model of UC-RP.

Proof: We assume that *ap* is a non-auxiliary path found in some route discovery of SMNDP, and $ap=(..., l_i, l_{i+1}, ..., l_{i+q}, l_{i+q+1}, ...)$, in which l_i and l_{i+q+1} are identifiers of non-adversarial nodes, they respectively correspond to non-adversarial vertices u and w in V, and $l_{i+1}, ..., l_{i+q}$ is a successive sequence of compromised identifiers used by the adversary A on an adversarial vertex v^* in V (Note that the neighbors of the adversarial vertex are non-adversarial vertices). To compute *ap*, the node that identifier is l_{i+q+1} receives the following route quest RREQ from the adversarial vertex v^* on the corresponding non-adversarial vertex w:

$$<ssid, RREQ, l_B, l_E, rp, ap= (...l_i, l_{i+1},...,l_{i+q}) >$$

When the node l_{i+q+1} receives this RREQ during this route discovery, for every reference path $path \in rp$, if:

- vertex v^* is present on a path

From the transmission strategy of the route request for computing the auxiliary path, we learn that the node l_{i+q+1} forwards or discards the RREQ according the location relationship of vertices u, v^* and w on the reference paths rather than directly according to the RREQ that the adversary forwarded to it on the vertex v^* in SMNDP. Therefore, if an adversarial vertex v^*, from which the adversary has forwarded the RREQ to vertex w, is present on a path in the reference paths *rp*, the node l_{i+q+1} can check out the faulty behavior that the adversary doesn't follow the requirement of SMNDP on the vertex v^*, and the node l_{i+q+1} discards the RREQ received from the vertex v^*.

As an example, let us consider the transmission rule 3 of SMNDP. We assume that vertices u and w are not present on any path of the reference paths *rp*, but the vertex v^* is present on a path rp_i of *rp*. The adversary forwards the RREQ to w on the vertex v^* when it receives the RREQ from u. The node l_{i+q+1} will discard the RREQ (the transmission rule 3 in table 2), because the node l_{i+q+1} finds that the adversary should unicast the

RREQ to the predecessor of v^* on the route rp_i instead of w according to the location relationship of vertices u, v^* and w on the reference paths rp. The error-check scheme in SMNDP checks out this attack.

So, the uniquely possible cause made *ap* a non-auxiliary path is that the adversary modifies the reference paths *rp* during the course of computing the path *ap*.

However, the source node *B* can check out this attack and delete the route reply message RREP according to the reference paths in route cache and the Message Authentication Code (MAC) that contains in route reply message RREP. The case that the adversary modifies *rp* can occur only with negligible probability if the message authentication scheme used in SMNDP is secure.

● the vertex v^* is not present on any path

In this situation, according to the algorithm and its own location relationship on the reference paths, the node l_{i+q+1} directly forwards the RREQ from the adversarial vertex v^*. To make the algorithm simple and efficient we do so, because without the help of the node l_{i-1}, the node l_{i+q+1} cannot determine whether the adversary has forwarded the RREQ according to the requirement of SMNDP. However, if the vertex v^* is not present on the reference path, even if the adversary doesn't follow the requirement of SMNDP, the reorganization of *ap* found in this route discovery and a plausible-route set in route cache cannot generate vertex-intersecting routes. Therefore, we still consider this *ap* as a auxiliary path in this case.

So, SMNDP returns a non-auxiliary path on any configuration *conf* only with negligible probability in ideal-world model of UC-RP.

Theorem 1: SMNDP is a UC-secure multiple node-disjoint paths source routing protocol in UC-RP, if the signature scheme and the message authentication scheme used by SMNDP are secure.

Proof: Firstly, we apply the simulator technology to design all kinds of stimulant scenarios, and then show that definition 3 can be satisfied. Let *A* is an active adversary, we construct a ideal process adversary *S* (a "simulator") for ideal functionality $F_{routing}$ in ideal process. In ideal process, *S* interacts with ideal functionality $F_{routing}$ and the environment *Z*. *S* is invoked by $\tilde{A}$ that it is a copy of *A*. $\tilde{A}$ interacts with real-model adversary *A*. The interaction of the simulator *S* is classified into outside interaction and inside interaction. Outside interaction is interactive behavior of the simulator *S* in ideal process, and inside interaction is interaction between the adversary *A* and $\tilde{A}$. The simulator *S* runs as follows:

(1) *S* simulates the first route discovery process:

Based on any configuration *conf*, there exists an active adversary *A*. The simulator *S* gets a message $<ssid=1, RREQ, l_B, l_E, rp=\Phi, ap_1>$ from real-model player M_i (*A* has compromised the player M_i) via a copy $\tilde{A}$ of *A* in inside-interaction simulation. In outside-interaction simulation, the simulator *S* mimics the behavior of real-model player M_i (*A* has compromised the player M_i) and gets a message $< ssid=1, RREP, l_B, l_E, rp=\Phi, ap_1, Sig_{l_E}, Sig_{l_n}, \ldots, Sig_{l_{i-1}} >$ that the ideal functionality $F_{routing}$ forwards to dummy player M'_i.

(2) *S* simulates the *x* route discovery process:

Based on any configuration *conf*, there exists an active adversary *A*. In inside-interaction simulator, the simulator *S* gets a message $<ssid=x, RREQ, l_B, l_E, rp, ap_x>$ from real-model player M_i via a copy $\tilde{A}$ of *A*. In outside-interaction simulation, the simulator *S* mimics the behavior of real-model player M_i and gets a message $< ssid=x, RREP, l_B, l_E, ap_x, MAC_{K_{B,E}}(ssid, rp), Sig_{l_E}, Sig_{l_n}, \ldots, Sig_{l_{i-1}} >$ that the ideal functionality $F_{routing}$ forwards to dummy player M'_i.

Now, all of states that the simulator *S* simulates route discovery process are described as above.

The proof of definition 3 is shown as follows. View that get from the interaction of the environment *Z*, the adversary *A* and the player in UC-RP and view that get from the interaction of the environment *Z*, the ideal-model adversary *S* and a copy $\tilde{A}$ in inside-interaction simulation of the simulator *S* is equal. So, the indistinguishable proof of these two views can be transformed into the indistinguishable proof of inside-interaction simulation and outside-interaction simulation of the simulator *S*. For any configuration $conf=(G(V, E), V^*, F)$ and any adversary *A*, SMNDP returns a error route (non-plausible route and non-auxiliary path) only with negligible probability in ideal model of UC-RP. Namely, the simulation of the simulator *S* is perfect. The indistinguishable proof can be illustrated by reduction to absurdity. Let us suppose the environment *Z* can distinguish the behavior of the simulator *S*, there exist the following two cases:

(1) *ap* found in each route discovery of SMNDP is a non-plausible route on any configuration *conf* only with negligible probability.

The signature scheme is used when the destination node and the intermediate node return a route reply message. So, if the signature scheme used in SMNDP is secure, *ap* found in this route discovery is a non-plausible route only with negligible probability in ideal-world model of UC-RP. The details of the proof are shown in the proof of lemma 1.

(2) *ap* found in this route discovery of SMNDP is a non-auxiliary path on any configuration *conf* only with negligible probability.

The Message Authentication Code (MAC) on the current reference paths is used when the destination node return a route reply message. So, if MAC used by SMNDP on the reference paths *rp* is secure, *ap* found in this route discovery is a non-auxiliary path only with negligible probability in ideal-world model of UC-RP. The details of the proof are shown in the proof of lemma 2.

From what is discussed above (1) and (2), the probability that SMNDP returns a error route (non-plausible route and non-auxiliary path) is negligible in ideal-world model of UC-RP. So, the reorganization at the source *S* generates a nonplausible-route set only with negligible probability. SMNDP is a provably secure multiple node-disjoint paths secure source routing protocol in UC-RP.■

6. Efficiency Analysis of Relative Solutions

Some of existing multi-path routing protocols (e.g. SDMSR[21], SecMR[22]) also address the security issues. However, the security of these protocols is analyzed by informal methods only or with formal methods that have never been intended for the analysis of this kind of protocol (e.g., BAN logic). In this paper, the adversary model (*active-n-m* attacker) firstly is introduced, and then UC-RP security framework for routing protocol and the security definition of multiple node-disjoint paths routing based on this framework are proposed. Finally, the security of SMNDP is discussed in UC-RP.

Existing multi-path routing protocols such as SDMSR, SecMR etc. has no any error-check scheme, however, the adversarial nodes in the network don't possibly follow the requirement of protocol when they forward the route request. The error-check scheme is used in the algorithm of route discovery for SMNDP. We extend the transmission strategy of the route request for two nodes in MNDP into the transmission strategy of the route request for three nodes. The scheme can check out the faulty behavior that the adversary violates the transmission strategy of the route request for SMNDP. In comparison with the algorithm of route discovery for MNDP, the error-check scheme only adds the computation cost that the intermediate node received the route request judges the location relationship of three relative nodes on the reference paths. But no any cryptography operation is used in this scheme.

In addition, besides being provably secure, SMNDP has another significant advantage over similar protocols (e.g. SDMSR, SecMR etc.): it is more efficient, because it requires less cryptographic computation overall from the nodes. This is because in SMNDP, only the processing of the route reply messages involves cryptographic

	MNDP	SMNDP	SDMSR	SecMR
Adversarial model	No	Active-n-m attack	No	No
Security model	No	UC-RP	No	No
Security definition	No	Yes	No	No
Error check	No	Yes	No	No
Cryptographic primitives	No	Signature，MAC	RSA-TC(n,2), Signature, MAC	Signature, Hash，Public key encryption
Number of cryptographic operations	No	$\|V'\|$ $+ k\sum_{i=1}^{\|V'\|/k+1} i + 4$ k ($\|V'\|<\|V\|$)	$\sum_{v\in V} deg(v) + \|V'\|$ $+ k\sum_{i=1}^{\|V'\|/k+1} i + 4k + 2$	$\|V\| + \sum_{v\in V} deg(v) + 2k + 4$
Number of route requests	at most $k(\|V\|-1)$	at most $k(\|V\|-1)$	at least $\|V\|$-1 At most $\max_{v\in V-\{B,E\}} deg(v)*(\|V\|-2)$ $+1-deg(E)$	at least $\|V\|$-1 At most $\max_{v\in V-\{B,E\}} deg(v)*(\|V\|-2)$ $+1-deg(E)$

Table 3. Comparison of multiple node-disjoint paths routing

operations, and a route reply message is processed only by those nodes that are carried in the route reply. In contrast to this, in SDMSR etc., the route request messages need to carry out cryptographic operations by all intermediate nodes; however, due to the way a route request is propagated, this means that each node in the network must involve cryptographic operations on each and every route request. So, SMNDP is more efficient than other similar multi-path secure protocols.

Let us suppose that V is a set of network nodes in mobile Ad hoc networks, and that $deg(v)$ is number of neighbor nodes of $v \in V$. We assume that B and E are respectively the source node and the destination node, and that V' is a set of nodes on all k node-disjoint paths between the source node B and the destination node E.

Communication overhead is a good indicator of both complexity and performance for an Ad hoc network routing protocol. For on-demand routing protocols, there are two components of communication overhead: unicast messages and broadcast messages. In multi-path routing protocols, the unicast overhead is proportional to the broadcast overhead and usually a small fraction of broadcast overhead. Hence, we consider only broadcast messages to compare the communication overhead of these protocols. Namely, we only compare the number that network nodes forward route request RREQ. In addition, we assume that the complexity of cryptographic primitives used in SMNDP, SDMSR and SecMR are same, so we only compare the number that these protocols perform cryptographic operations.

In SMNDP, the destination node and the intermediate nodes sign on the route reply RREP, the route discovery. So, the number of cryptographic operations for signature and signature verification is $|V'|+k\sum_{i=1}^{|V'|/k+1} i + 2k$. In addition, the destination node and the source node and the intermediate nodes verify signature when the destination node and the intermediate nodes return the path information found in generates a Message Authentication Code (MAC), and the source node verifies the MAC in every route discovery. The number of cryptographic operations for MAC is $2k$. Thus, the cryptographic operations that SMNDP performs to complete the route discovery process is V' $|+k\sum_{i=1}^{|V'|/k+1} i + 4k$ $(|V'|<|V|)$.

SDMSR is similar to SMNDP, the number of cryptographic operations that the destination node and the intermediate nodes sign on the route reply RREP and verify signatures is $|V'|+k\sum_{i=1}^{|V'|/k+1} i + 4k$. In addition, the route request RREQ is authenticated by the destination node and the intermediate nodes that received the RREQ, and the intermediate nodes forward every route request that they receive (we assume that any RREQ with a smaller path than the precedent one). So, the number of cryptographic operations in route discovery process is $\sum_{v\in V} deg(v) + 2$. Thus, the cryptographic operations that SDMSR performs to complete the route discovery process is $\sum_{v\in V} deg(v) + |V'| + k\sum_{i=1}^{|V'|/k+1} i + 4k + 2$.

SecMR mainly performs the cryptographic operations in neighborhood authentication phase. To complete the mutual authentication of network nodes, in periodic time intervals, each node broadcasts to its one-hop neighbors a signed message including the current time and its unique identifier. Thus, each node generates one signature and verifies one signature for each one of its neighbors. So, the number of cryptographic operations in neighborhood authentication phase is $|V| + \sum_{v\in V} deg(v)$. In route request message, there are an encryption operation and a hash operation at the source node. Correspondingly, the destination node performs decryption operations and hash operation when it receives the RREQ that come from the source node. In addition, the destination node performs hash operations on each path when it returns the path information, so, the source node reads these hash values when it receives these route-reply messages. Thus, the cryptographic operations that SecMR performs to complete the route discovery process is $|V| + \sum_{v\in V} deg(v) + 2k + 4$.

SMNDP and MNDP identify the maximal set of node-disjoint paths in multiple route discoveries and in an incremental fashion. In each route discovery of SMNDP and MNDP, the intermediate nodes only forward the first route request RREQ. So, in k route discovery, network nodes forward $k(|V|\text{-}1)$ route request RREQs to compute k node-disjoint paths.

SDMSR and SecMR compute the maximal set of node-disjoint paths in single route discovery. Either at the source node or at the destination node identifies a maximal set of node-disjoint paths from a list of paths traversed by different copies of an RREQ. They require that the intermediate nodes forward all of RREQs received by these nodes. Thus, the number of route request RREQ that SDMSR and SecMR forwards is at least $|V|$-1 and at most $\max_{v\in V-\{B,E\}} deg(v)*(|V|-2)+1-deg(E)$. The result of comparing is listed in table 3.

7. Conclusion and Future Work

It is a challenge to identify node-disjoint paths in mobile Ad hoc networks. In this paper, we propose a multiple node-disjoint paths secure source routing protocol SMNDP based on MNDP protocol. In contrast to MNDP, the error-check and cryptographic mechanisms are used in SMNDP. In this paper, to the best of our knowledge, we firstly apply the notions of UC security framework in the context of routing protocols for MANET to analyze the security of routing protocol, and propose UC-RP security framework and an ideal functionality $F_{Routing}$ that capture the security requirement of routing protocol. And then, we prove that SMNDP satisfies the security definition in UC-RP. UC-RP security framework can analyze composably secure property of routing protocol in complicated and unpredictable communication network environment, so it is suitable for mobile Ad hoc networks applications.

In this paper, we mainly focus on on-demand multi-path source routing protocols, but similar principles can be applied to other types of protocols too (e.g. Ad hoc on-demand multi-path distance vector routing, AOMDV etc.). In addition, we note that SMNDP can be optimized with respect to communication overhead by replacing the signature list in the route reply with a single aggregate signature (e.g. [23]) computed by the intermediate nodes iteratively. The details of this optimization and its security analysis are left for future work.

8. References

[1] David B. Johnson and David A. Maltz. Dynamic Source Routing in Ad Hoc Wireless Networks . *Mobile Computing*, 1996, 12(6): 10-23.

[2] C. Perkins and E. Royer. Ad hoc on-demand distance vector routing. *Mobile Systems and Applications*. 1999, 24(3): 59-81.

[3] T. Clausen and P. Jacquet, “Optimized Link State Routing Protocol (OLSR),” Internet Request for Comments 3626, Oct. 2003.

[4] A.M.Abbs,p.khandput,B.N.Jain, NDMA: A Node Disjoint Multipath Ad hoc routing Protocol, *In proceedings of 5th World Wireless Congress(WWC)*, San Fransisco, IEEE Press, 2004,334-339.

[5] Z. Ye, S.V. Krishnamurthy, S.K. Tripathi, A Framework for Reliable Routing in Mobile Ad hoc Networks, *In Proceedings of in IEEE Conference on Computer and Communication (INFOCOM)*, San Fransisco, IEEE Press,2003, 270-280.

[6] P.P. Pham, S. Perreau. Performance Analysis of Reactive Shortest Path and Multi-path Routing Mechanism with Load Balance. *In Proceedings of IEEE Conference on Computer and Communication (INFOCOM)*, San Fransisco, IEEE Press, 2003, 251-259.

[7] Changwen Liu, Mark Yarvis, W.steven Conner, Xingang Guo. Guaranteed On-Demand Discovery of Node-Disjoint Paths in Ad hoc Networks, *Computer communications*, 2007, Vol 30:2917-2930.

[8] Ash Mohammad Abbas and Tehzeeb Ahmed Abbasi. An Improvement over Incremental Approach for Guaranteed Identification of Multiple Node-Disjoint Paths in Mobile Ad hoc Networks, *In Communication Systems Software and Middleware, 2nd International Conference* , Bangalore, IEEE Press, 2007,1-10.

[9] Y.-C. Hu and A. Perrig.A Survey of Secure Wireless Ad Hoc Routing, *IEEE Security and Privacy Magazine*, 2004 , Vol 2(3), 28-39.

[10] J. Marshall. An Analysis of the Secure Routing Protocol for Mobile Ad Hoc Network Route Discovery: Using Intuitive Reasoning and Formal Verification to Identify Flaws. MSc thesis, Dept. of Computer Science, Florida State Univ., Apr. 2003.

[11] S. Yang and J. Baras. Modeling Vulnerabilities of Ad Hoc Routing Protocols. *Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks*, Fairfax, Virginia, ACM Press, 2003.12-20.

[12] G. Acs, L. Buttyan, and I. Vajda. Provably secure on-demand source routing in mobile ad hoc networks. *Technical Report 159, International Association for Cryptologic Research*, 2004.

[13] Gergely Acs, Levente Buttyan, and Istvan Vajda. Provably secure on-demand source routing in mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, 2006 , Vol 5(11):1533-1546.

[14] B. Pfitzmann and M. Waidner. A Model for Asynchronous Reactive Systems and Its Application to Secure Message Transmission. *In IEEE Symposium on Security and Privacy*, IEEE Press, 2001,184-200.

[15] Tao Feng, Xian Guo, Jianfeng Ma, SMNDP: A Practical and Provably Secure Multi-path Source Routing, *In proceeding of The 4th International Conference on Mobile Ad-hoc and Sensor Networks (MSN 2008),* IEEE press, 2008, 63-67.

[16] R.Canetti. Universally Composable Security: A New Paradigm for Cryptographic Protocols. *In proceedings of the 42nd IEEE symposium on the FOCS. New York* , IEEE Computer Society Press,2001, 136-145.

[17] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*, Second Edition, Cambridge, MA,The MIT Press, 1993.

[18] J.R. Douceur. The Sybil Attack. *In Proceedings of First International Workshop Peer-to-Peer Systems (IPTPS)*, Cambridge, MA, Springer Berlin Press, 2002, 251-260.

[19] Y.-C. Hu, A. Perrig, and D. Johnson. Packet Leashes: A Defense against Wormhole Attacks in Wireless Ad Hoc Networks. *In Proceedings of the Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003)*. San Francisco, CA, IEEE Press, 2003, vol. 3, 1976-1986.

[20] Feng Tao, Ma Jianfeng, A General Key Seed Management and Assignment Model for Wireless Sensor networks and Application, *Journal of Computer Research And Development*,2008, Vol 45(1): 146-153. (in chinese)

[21] Sebastien Berton, Hao Yin and Chuang Lin,Ge Yong-Min,Secure, Disjoint, Multipath Source Routing Protocol (SDMSR) for Mobile Ad-Hoc Networks, *In Proceedings of the Fifth International Conference on Grid and Cooperative Computing (GCC'06)*, Washington DC, IEEE Computer Society Press, 2006,387-394.

[22] Panayiotis Kotzanikolaou,Rosa Mavropodi,Christos Douligeris. Secure Multipath Routing for Mobile Ad hoc Networks,*Ad hoc networks*, 2007,Vol 5(1): 87-99

[23] D. Boneh, C. Gentry, H. Shacham, and B. Lynn, Aggregate and Verifiably Encrypted Signatures from Bilinear Maps, *Advances in Cryptology—Proc. Eurocrypt '03*, 2003.