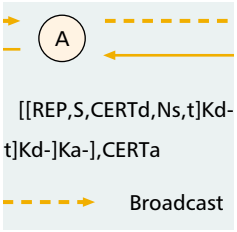
SECURITY IN MOBILE AD HOC NETWORKS: CHALLENGES AND SOLUTIONS

HAO YANG, HAIYUN LUO, FAN YE, SONGWU LU, AND LIXIA ZHANG,
UCLA COMPUTER SCIENCE DEPARTMENT



Security has become a primary concern to provide protected communication between mobile nodes in a hostile environment. Unlike wireline networks, the unique characteristics of mobile ad hoc networks pose a number of non-trivial challenges to the security design.

ABSTRACT

Security has become a primary concern in order to provide protected communication between mobile nodes in a hostile environment. Unlike the wireline networks, the unique characteristics of mobile ad hoc networks pose a number of nontrivial challenges to security design, such as open peer-to-peer network architecture, shared wireless medium, stringent resource constraints, and highly dynamic network topology. These challenges clearly make a case for building multifence security solutions that achieve both broad protection and desirable network performance. In this article we focus on the fundamental security problem of protecting the multihop network connectivity between mobile nodes in a MANET. We identify the security issues related to this problem, discuss the challenges to security design, and review the state-of-the-art security proposals that protect the MANET link- and network-layer operations of delivering packets over the multihop wireless channel. The complete security solution should span both layers, and encompass all three security components of prevention, detection, and reaction.

INTRODUCTION

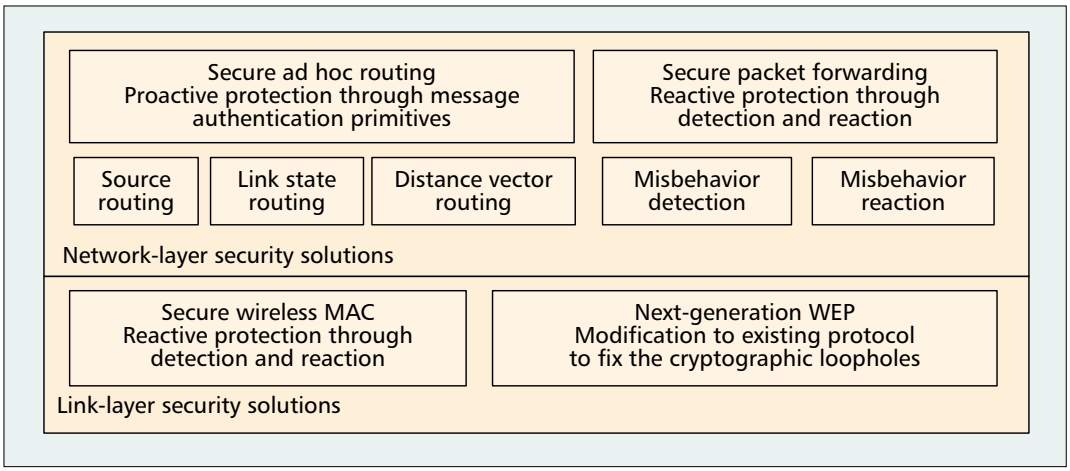
In recent years mobile ad hoc networks (MANETs) have received tremendous attention because of their self-configuration and self-maintenance capabilities. While early research effort assumed a friendly and cooperative environment and focused on problems such as wireless channel access and multihop routing, security has become a primary concern in order to provide protected communication between nodes in a potentially hostile environment. Although security has long been an active research topic in wireline networks, the unique characteristics of MANETs present a new set of nontrivial challenges to security design. These challenges include open network architecture, shared wireless medium, stringent resource constraints, and highly dynamic network topology. Consequently, the existing security solutions for wired networks do not directly apply to the MANET domain.

The ultimate goal of the security solutions for MANETs is to provide security services, such as authentication, confidentiality, integrity, anonymity, and availability, to mobile users. In order to achieve this goal, the security solution should provide complete protection spanning the entire protocol stack. Table 1 describes the security issues in each layer. In this article we consider a fundamental security problem in MANET: *the protection of its basic functionality to deliver data bits from one node to another*. In other words, we seek to protect the network connectivity between mobile nodes over potentially multihop wireless channels, which is the basis to support any network security services. Multihop connectivity is provided in MANETs through two steps: (1) ensuring one-hop connectivity through link-layer protocols (e.g., wireless medium access control, MAC); and (2) extending connectivity to multiple hops through network-layer routing and data forwarding protocols (e.g., ad hoc routing). Accordingly, we focus on the link- and network-layer security issues, challenges, and solutions in MANETs in this article.

One distinguishing characteristic of MANETs from the security design perspective is the lack of a *clear* line of defense. Unlike wired networks that have dedicated routers, each mobile node in an ad hoc network may function as a router and forward packets for other peer nodes. The wireless channel is accessible to both legitimate network users and malicious attackers. There is no well defined place where traffic monitoring or access control mechanisms can be deployed. As a result, the boundary that separates the inside network from the outside world becomes blurred. On the other hand, the existing ad hoc routing protocols, such as Ad Hoc On Demand Distance Vector (AODV) [1] and Dynamic Source Routing (DSR) [2], and wireless MAC protocols, such as 802.11 [3], typically assume a trusted and cooperative environment. As a result, a malicious attacker can readily become a router and disrupt network operations by intentionally disobeying the protocol specifications.

There are basically two approaches to protecting MANETs: proactive and reactive. The proactive approach attempts to prevent an attacker from launching attacks in the first place, typically through various cryptographic techniques. In

1536-1284/04/$20.00 © 2004 IEEE



■ Figure 1. *The components in the multifence security solution.*

contrast, the reactive approach seeks to detect security threats *a posteriori* and react accordingly. Due to the absence of a clear line of defense, a complete security solution for MANETs should integrate both approaches and encompass all three components: prevention, detection, and reaction. For example, the proactive approach can be used to ensure the correctness of routing states, while the reactive approach can be used to protect packet forwarding operations. As argued in [4], security is a chain, and it is only as secure as the weakest link. Missing a single component may significantly degrade the strength of the overall security solution.

Security never comes for free. When more security features are introduced into the network, in parallel with the enhanced security strength is the ever-increasing computation, communication, and management overhead. Consequently, network performance, in terms of scalability, service availability, robustness, and so on of the security solutions, becomes an important concern in a resource-constrained ad hoc network. While many contemporary proposals focus on the security vigor of their solutions from the cryptographic standpoint, they leave the network performance aspect largely unaddressed. In fact, both dimensions of security strength and network performance are equally important, and achieving a good trade-off between two extremes is one fundamental challenge in security design for MANETs.

This article is structured as follows. We describe the attack model in the next section, and then identify the challenges in MANET security design. Next, we overview the state-of-the-art security proposals that protect MANET from different types of attacks in the link and network layers, respectively. Lastly, we discuss open challenges and possible future directions in this area.

Attacks

A MANET provides network connectivity between mobile nodes over potentially multihop wireless channels mainly through link-layer protocols that ensure one-hop connectivity, and network-layer protocols that extend the connectivity

Layer	Security issues
Application layer	Detecting and preventing viruses, worms, malicious codes, and application abuses
Transport layer	Authenticating and securing end-to-end communications through data encryption
Network layer	Protecting the ad hoc routing and forwarding protocols
Link layer	Protecting the wireless MAC protocol and providing link-layer security support
Physical layer	Preventing signal jamming denial-of-service attacks

■ **Table 1.** *The security solutions for MANETs should provide complete protection spanning the entire protocol stack.*

to multiple hops. These distributed protocols typically assume that all nodes are cooperative in the coordination process. This assumption is unfortunately not true in a hostile environment. Because cooperation is assumed but not enforced in MANETs, malicious attackers can easily disrupt network operations by violating protocol specifications.

The main network-layer operations in MANETs are ad hoc routing and data packet forwarding, which interact with each other and fulfill the functionality of delivering packets from the source to the destination. The ad hoc routing protocols exchange routing messages between nodes and maintain routing states at each node accordingly. Based on the routing states, data packets are forwarded by intermediate nodes along an established route to the destination. Nevertheless, both routing and packet forwarding operations are vulnerable to malicious attacks, leading to various types of malfunction in the network layer. While a comprehensive enumeration of the attacks is out of our scope, such network-layer vulnerabilities generally fall into one of two categories: *routing attacks* and *packet forwarding attacks*, based on the target operation of the attacks.

The family of routing attacks refers to any action of advertising routing updates that does not follow the specifications of the routing protocol. The specific attack behaviors are related

Unlike wired networks that have dedicated routers, each mobile node in an ad hoc network may function as a router and forward packets for other nodes. The wireless channel is accessible to both legitimate network users and malicious attackers.

to the routing protocol used by the MANET. For example, in the context of DSR [2], the attacker may modify the source route listed in the RREQ or RREP packets by deleting a node from the list, switching the order of nodes in the list, or appending a new node into the list [5]. When distance-vector routing protocols such as AODV [1] are used, the attacker may advertise a route with a smaller distance metric than its actual distance to the destination, or advertise routing updates with a large sequence number and invalidate all the routing updates from other nodes [6]. By attacking the routing protocols, the attackers can attract traffic toward certain destinations in the nodes under their control, and cause the packets to be forwarded along a route that is not optimal or even nonexistent. The attackers can create routing loops in the network, and introduce severe network congestion and channel contention in certain areas. Multiple colluding attackers may even prevent a source node from finding any route to the destination, and partition the network in the worst case.

There are still active research efforts in identifying and defeating more sophisticated and subtle routing attacks. For example, the attacker may further subvert existing nodes in the network, or fabricate its identity and impersonate another legitimate node [7]. A pair of attacker nodes may create a wormhole [8] and shortcut the normal flows between each other. In the context of on-demand ad hoc routing protocols, the attackers may target the route maintenance process and advertise that an operational link is broken [5].

In addition to routing attacks, the adversary may launch attacks against packet forwarding operations as well. Such attacks do not disrupt the routing protocol and poison the routing states at each node. Instead, they cause the data packets to be delivered in a way that is intentionally inconsistent with the routing states. For example, the attacker along an established route may drop the packets, modify the content of the packets, or duplicate the packets it has already forwarded. Another type of packet forwarding attack is the denial-of-service (DoS) attack via network-layer packet blasting, in which the attacker injects a large amount of junk packets into the network. These packets waste a significant portion of the network resources, and introduce severe wireless channel contention and network congestion in the MANET.

Recent research efforts have also identified the vulnerabilities of the link-layer protocols, especially the de facto standard IEEE 802.11 MAC protocol [3], for MANETs. It is well known that 802.11 WEP is vulnerable to several types of cryptography attacks due to the misuse of the cryptographic primitives [9]. The 802.11 protocol is also vulnerable to DoS attacks targeting its channel contention and reservation schemes. The attacker may exploit its binary exponential backoff scheme to deny access to the wireless channel from its local neighbors [10, 11]. Because the last winner is always favored among local contending nodes, a continuously transmitting node can always capture the channel and cause other nodes to back off endlessly. Moreover, backoffs at the link layer can incur a chain reaction in upper layer protocols using backoff schemes (e.g., TCP's window management). Another vulnerability of 802.11 comes from the NAV field carried in the request to send/clear to send (RTS/CTS) frames, which indicates the duration of channel reservation. An adversarial neighbor of either the sender or the receiver may overhear the NAV information and then intentionally introduce a 1-bit error into the victim's link-layer frame by wireless interference. The corrupted frame has to be discarded by the receiver after error detection. This effectively constitutes another type of DoS attack.

CHALLENGES

One fundamental vulnerability of MANETs comes from their open peer-to-peer architecture. Unlike wired networks that have dedicated routers, each mobile node in an ad hoc network may function as a router and forward packets for other nodes. The wireless channel is accessible to both legitimate network users and malicious attackers. As a result, there is no clear line of defense in MANETs from the security design perspective. The boundary that separates the inside network from the outside world becomes blurred. There is no well defined place/infrastructure where we may deploy a single security solution.

Moreover, portable devices, as well as the system security information they store, are vulnerable to compromises or physical capture, especially low-end devices with weak protection. Attackers may sneak into the network through these subverted nodes, which pose the weakest link and incur a domino effect of security breaches in the system.

The stringent resource constraints in MANETs constitute another nontrivial challenge to security design. The wireless channel is bandwidth-constrained and shared among multiple networking entities. The computation capability of a mobile node is also constrained. For example, some low-end devices, such as PDAs, can hardly perform computation-intensive tasks like asymmetric cryptographic computation. Because mobile devices are typically powered by batteries, they may have very limited energy resources.

The wireless medium and node mobility poses far more dynamics in MANETs compared to the wireline networks. The network topology is highly dynamic as nodes frequently join or leave the network, and roam in the network on their own will. The wireless channel is also subject to interferences and errors, exhibiting volatile characteristics in terms of bandwidth and delay. Despite such dynamics, mobile users may request for anytime, anywhere security services as they move from one place to another.

The above characteristics of MANETs clearly make a case for *building multifence security solutions that achieve both broad protection and desirable network performance*. First, the security solution should spread across many individual components and rely on their collective protection power to secure the entire network. The security scheme adopted by each device has to work within its own resource limitations in terms

of computation capability, memory, communication capacity, and energy supply. Second, the security solution should span different layers of the protocol stack, with each layer contributing to a line of defense. No single-layer solution is possible to thwart all potential attacks. Third, the security solution should thwart threats from both outsiders who launch attacks on the wireless channel and network topology, and insiders who sneak into the system through compromised devices and gain access to certain system knowledge. Fourth, the security solution should encompass all three components of prevention, detection, and reaction, that work in concert to guard the system from collapse. Last but not least, the security solution should be practical and affordable in a highly dynamic and resource-constrained networking scenario.

A Multifence Security Solution

In this section we review the state-of-the-art security proposals for MANETs. Because multihop connectivity is provided in MANETs through distributed protocols in both the network and link layers, the ultimate multifence security solution naturally spans both layers, as illustrated in Fig. 1.

There are basically two approaches to securing a MANET: proactive and reactive. The proactive approach attempts to thwart security threats in the first place, typically through various cryptographic techniques. On the other hand, the reactive approach seeks to detect threats *a posteriori* and react accordingly. Each approach has its own merits and is suitable for addressing different issues in the entire domain. For example, most secure routing protocols adopt the proactive approach in order to secure routing messages exchanged between mobile nodes, while the reactive approach is widely used to protect packet forwarding operations.

Due to the absence of a clear line of defense, a complete security solution for MANETs should integrate both proactive and reactive approaches, and encompass all three components: prevention, detection, and reaction. The prevention component deters the attacker by significantly increasing the difficulty of penetrating the system. However, the history of security has clearly shown that a completely intrusion-free system is infeasible, no matter how carefully the prevention mechanisms are designed. This is especially true in MANETs, consisting of mobile devices that are prone to compromise or physical capture. Therefore, the detection and reaction components that discover the occasional intrusions and take reactions to avoid persistent adverse effects, are indispensable for the security solutions to operate in the presence of limited intrusions.

In the MANET context, the prevention component is mainly achieved by secure ad hoc routing protocols that prevent the attacker from installing incorrect routing states at other nodes. These protocols are typically based on earlier ad hoc routing protocols such as DSR [2], AODV [1], and Destination-Sequenced Distance Vector (DSDV) [12], and employ different cryptographic primitives (e.g., HMAC, digital signatures, hash chains) to authenticate the routing messages. The detection component discovers ongoing attacks through identification of abnormal behavior exhibited by malicious nodes. Such misbehavior is detected either in an end-to-end manner, or by the neighboring nodes through overhearing the channel and reaching collaborative consensus. Once an attacker node is detected, the reaction component makes adjustments in routing and forwarding operations, ranging from avoiding the node in route selection to collectively excluding the node from the network.

Network-Layer Security

The network-layer security designs for MANETs are concerned with protecting the network functionality to deliver packets between mobile nodes through multihop ad hoc forwarding. Therefore, they seek to ensure that the routing message exchanged between nodes is consistent with the protocol specification, and the packet forwarding behavior of each node is consistent with its routing states. Accordingly, the existing proposals can be classified into two categories: *secure ad hoc routing protocols* and *secure packet forwarding protocols*. Before we describe these security solutions in detail, we first introduce several cryptographic primitives for message authentication, the essential component in any security design, and analyze the trade-offs behind them.

Message Authentication Primitives — There are three cryptographic primitives widely used to authenticate the content of messages exchanged among nodes.

HMAC (message authentication codes).[1] If two nodes share a secret symmetric key K, they can efficiently generate and verify a message authenticator $h_K(\cdot)$ using a cryptographic one-way hash function h. The computation is very efficient, even affordable for low-end devices such as small sensor nodes. However, an HMAC can be verified only by the intended receiver, making it unappealing for broadcast message authentication. Besides, establishing the secret key between any two nodes is a nontrivial problem. If the pairwise shared key is used, a total number of

$$\frac{n \cdot (n-1)}{2}$$

keys will be maintained in a network with n nodes. SRP for DSR [13] takes this approach with pairwise shared keys.

Digital signature. Digital signature is based on asymmetric key cryptography (e.g., RSA), which involves much more computation overhead in signing/decrypting and verifying/encrypting operations. It is less resilient against DoS attacks since an attacker may feed a victim node with a large number of bogus signatures to exhaust the victim's computation resources for verifying them. Each node also needs to keep a certificate revocation list (CRL) of revoked certificates. However, a digital signature can be verified by any node given that it knows the public key of the signing node. This makes digital signature scalable to large numbers of receivers. Only a total number of n public/private key pairs

Due to the absence of a clear line of defense, a complete security solution for MANETs should integrate both proactive and reactive approaches, and encompass all three components: prevention, detection, and reaction.

[1] *In network literature, MAC normally refers to the medium access control protocol at the link layer. To avoid ambiguity, we use MAC to refer to link-layer medium access control, and HMAC to refer to keyed hashing for message authentication.*

S : $p_S = (RREQ,S,D)$, $m_S = HMAC_{K_{SD}}(p_S)$
$S \rightarrow *$: (p_S, m_S)
A : $h_A = H(A, m_S)$, $p_A = (RREQ,S,D,[A],h_A,[])$, $m_A = HMAC_{K_A}(p_A)$
$A \rightarrow *$: (p_A, m_A)
B : $h_B = H(B, h_A)$, $p_B = (RREQ,S,D,[A,B],h_B,[m_A])$, $m_B = HMAC_{K_B}(p_B)$
$B \rightarrow *$: (pB, mB)
C : $h_C = H(C, h_B)$, $p_C = (RREQ,S,D,[A,B,C],h_C,[m_A,m_B])$, $m_C = HMAC_{K_C}(p_C)$
$C \rightarrow *$: (p_C, m_C)
D : $p_D = (RREP,D,S,[A,B,C],[m_A,m_B,m_C])$, $m_D = HMAC_{K_{DS}}(p_D)$
$D \rightarrow C$: $(p_D, m_D, [])$
$C \rightarrow B$: $(p_D, m_D, [K_C])$
$B \rightarrow A$: $(p_D, m_D, [K_C, K_B])$
$A \rightarrow S$: $(p_D, m_D, [K_C, K_B, K_A])$

Figure 2. *The sequence of secure routing message exchange in Ariadne.*

need be maintained in a network of n nodes. SAODV [6] and ARAN [7] take the digital signature approach.

One-way HMAC key chain. Many cryptographic one-way functions exist such that given the output $f(x)$, it is computationally infeasible to find the input x. By applying $f(\cdot)$ repeatedly on an initial input x, one can obtain a chain of outputs $f^i(x)$. These outputs can be used in the reverse order of generation to authenticate messages: a message with an HMAC using $f^i(x)$ as the key is proven to be authentic when the sender reveals $f^{i-1}(x)$. TESLA [14] is one such hash-chain-based protocol commonly used to authenticate broadcast messages. SEAD for DSDV [15], Ariadne for DSR [5], and packet leashes [8] for wormhole attacks all take this approach.

The computation involved in one-way key-chain-based authentication is lightweight, and one authenticator can be verified by large numbers of receivers. However, these benefits come at a certain cost. First, hash-chain-based authentication requires clock synchronization at granularities that may need special hardware support. Second, receivers need to buffer a message to verify them when the key is revealed. The delay in the verification of routing messages may greatly decrease the responsiveness of the routing protocol. If immediate authentication is desired, very tight clock synchronization and large storage are necessary (e.g., TIK [8]). Third, the release of the key involves a second round of communication. The timer has to be carefully gauged according to the specific context. Finally, the storage of the hash chain is nontrivial for long chains, as required by scenarios with large rekeying intervals.

Secure Ad Hoc Routing

The secure ad hoc routing protocols take the proactive approach and enhance the existing ad hoc routing protocols, such as DSR and AODV, with security extensions. In these protocols, each mobile node proactively signs its routing messages using the cryptographic authentication primitives described above. This way, collaborative nodes can efficiently authenticate the legitimate traffic and differentiate the unauthenticated packets from outsider attackers. However, an authenticated node may have been compromised and controlled by the attacker. Therefore, we have to further ensure proper compliance with the routing protocols even for an authenticated node. In the following, we describe how different types of routing protocols are secured.

Source Routing — For source routing protocols such as DSR, the main challenge is to ensure that each intermediate node cannot remove existing nodes from or add extra nodes to the route. The basic technique is to attach a per-hop authenticator for the source routing forwarder list so that any altering of the list can be immediately detected (or after the key is disclosed for HMAC key-chain-based authentication).

A secure extension of DSR is Ariadne [5]. It uses a one-way HMAC key chain (i.e., TESLA) for the purpose of message authentication. Through key management and distribution, a receiver is assumed to possess the last released key of the sender's TESLA key chain. Take the following example for an illustration. The source node S uses source routing to connect to the destination D through three intermediate nodes A, B, and C. The protocol establishes a *hash chain* at the destination,

$$H(C, H(B, H(A, HMAC_{K_{SD}}(S,D)))),$$

where $HMAC_{K_{SD}}(M)$ denotes message M's HMAC code generated by a key shared between S and D. The well-known one-way hash function H authenticates the contents in the chain, and $HMAC_{K_{SD}}(S,D)$ authenticates the source-destination relation. The propagation of the route request (RREQ) and route reply (RREP) messages is described in Fig. 2, where * denotes a local broadcast and $HMAC_{K_X}(\cdot)$ denotes HMAC code generated on node X.

At the destination, D can compute m_S because information of p_S is contained in p_C. D dynamically computes h_C's value according to the explicit node list embedded in p_C, then compares this h_C to the one embedded in p_C for forgery detection. At the RREP phase, there is no need to generate separate authentication code for every RREP packet. By trapdoor commitment, any forwarder X already committed the one-way function outputs $m_X = HMAC_{K_X}(\cdot)$ at the RREQ phase; then at the RREP phase the commitment $m_X \rightarrow K_X$ is fulfilled by revealing key K_X.

Distance Vector Routing — For distance vector routing protocols such as DSDV and AODV, the main challenge is that each intermediate node has to advertise the routing metric correctly. For example, when hop count is used as the routing metric, each node has to increase the hop count by one exactly. A hop count hash chain [6, 15] is devised so that an intermediate node cannot *decrease* the hop count in a routing update. Note that a hash chain for this purpose does not need time synchronization, which is different from one-way HMAC key chain for authentication.

Assuming the maximum hop count of a valid route is n, a node generates a hash chain of length n every time it initiates an RREP message,

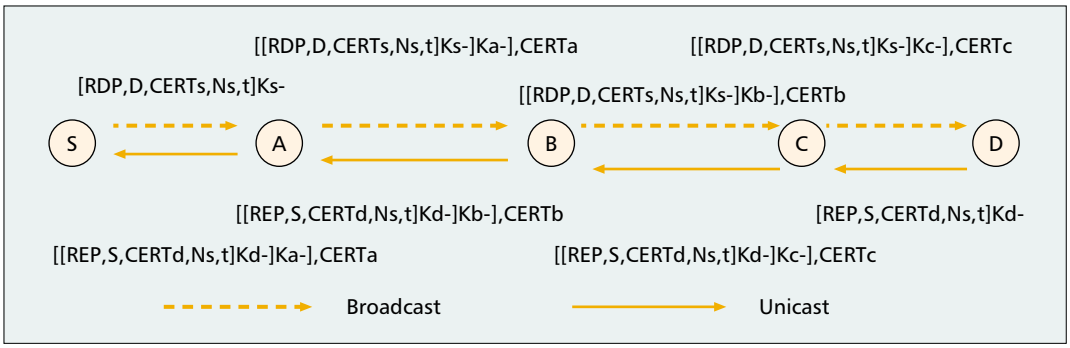


Figure 3. *The sequence of secure routing message exchange in ARAN.*

$h_0, h_1, h_2 \ldots, h_n$,

where $h_i = H(h_{i-1})$ and $H(\cdot)$ is a well-known one-way hash function. The node then adds $h_x = h_0$ and h_n into the routing message, with Hop_Count set to 0. Note that h_n and Hop_Count are authenticated with an authenticator according to the adopted authentication strategy discussed at the beginning of this section.

When a node receives an RREQ or RREP packet, it first checks whether

$h_n = H^{n-\text{Hop\_Count}}(h_x)$,

where $H_m(h_0)$ denotes the result of applying $H(\cdot)$ m times on h_x.

Then the node sets

$h_x = H(h_x)$.

Finally, the node increments the Hop_Count by 1, updates the authenticator, and forwards the route discovery packet.

This approach provides authentication for the lower bound of tbe hop count, but does not prevent a forwarder from advertising the *same* hop count as the one from another forwarder. In [8], a more complicated mechanism called a hash tree chain is proposed to ensure a monotonically increasing hop count as the routing update traverses the network. One general limitation of the above approaches is that they can only be used to protect discrete metrics. For continual metrics that take noninteger values, the one-way chain is ineffective.

Link State Routing — Secure Link State Routing (SLSP) [16] is a link state routing protocol for ad hoc networks. Its operations are similar to Internet link state routing protocols (e.g., Open Shortest Path First, OSPF): each node seeks to learn and update its neighborhood by *Neighbor Lookup Protocol* (NLP) and periodically floods *Link State Update* (LSU) packets to propagate link state information. NLP is responsible for:

- Maintaining mappings between MAC and IP addresses of a node's neighbors
- Identifying potential discrepancies, such as the use of multiple IP addresses on a single link
- Measuring the control packet rates from each neighbor

Neighbors use one-hop hello messages to discover each other, and connectivity is assumed to be lost if a hello message is not received within a timeout.

A node collects LSUs from all over the network in order to construct the global topology and calculate the route to any destination. Based on NLP, one LSU packet is constructed for each neighbor. Each LSU packet contains a sequence number and a hop count. Like DSR and AODV, duplicate LSU packets with previously seen sequence numbers are suppressed. The hop count determines the packet's time to live so that an LSU packet only travels within a zone, as in hybrid routing protocols like ZRP. An LSU receiving node adds a link to its global topology only if two valid LSUs from both nodes of the link are received. Thus, one malicious node alone cannot inject false link information successfully.

SLSP adopts a digital signature approach in authentication. NLP's hello messages and LSU packets are signed with the sender's private key. Any verifier can use the public key vouched for by the sender's valid certificate to verify a message's veracity. A certificate can be delivered to verifiers by either attachment to an LSU packet or dedicated *public key distribution* (PKD) packets. SLSP also employs various rate control mechanisms, such as time to live and rate throttle, in its NLP/LSU/PKD components. Thus, SLSP is less vulnerable to DoS attacks.

Other Routing Protocols — ARAN [7] ensures that each node knows the correct next hop on a route to the destination by public key cryptography. We illustrate the message exchange in ARAN using a simple example shown in Fig. 3. Each message is signed, and the sender's certificate is attached to prove the authenticity of its public keys. A source S floods the network with a signed RREQ packet. Upon receiving the first copy of RREQ, a node sets up state of a reverse path, pointing to the node from which it receives the RREQ. It then signs and broadcasts the packet. Upon receiving the RREQ, the destination D signs an RREP and unicasts it back on the reverse path. Each node along the reverse path signs the RREP and sends it to the next hop, which verifies the signature of the previous hop, until S receives the RREP. Thus, the discovered path is the one along which the first copy of RREQ reaches D from S; each node on this path knows the correct next hop, but not the whole path. It does not use any metric such as hop count, so the discovered path may not be optimal.

For distance vector routing protocols such as DSDV and AODV, the main challenge is that each intermediate node has to advertise the routing metric correctly. For example, when hop count is used as the routing metric, each node has to increase the hop count by one exactly.

A malicious node is possible to correctly participate in the route discovery phase but fails to correctly forward data packets. The security solution should ensure that each node indeed forwards packets according to its routing table.

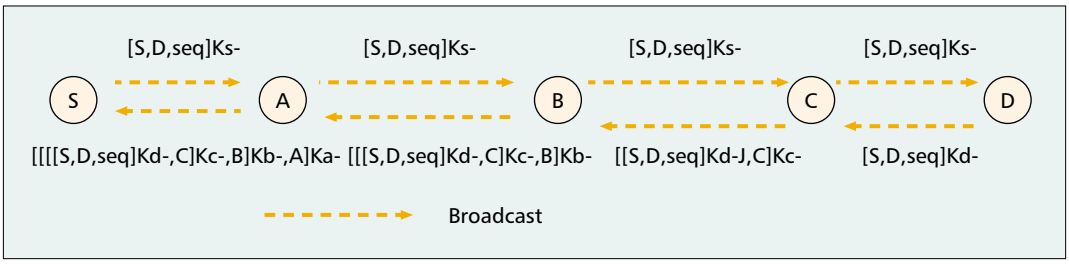


Figure 4. *The sequence of secure routing message exchange in a Byzantine-resilient routing protocol.*

Reference [17] proposes to flood both route requests and route replies in order to defend against Byzantine failures. When a source *S* needs a route to a destination *D*, it signs and floods an RREQ throughout the network, as shown in Fig. 4. When *D* receives the first copy of the request, it signs and floods an RREP that carries a route list so each intermediate hop can append its identifier. When a node receives the reply, it computes the total cost of the path as contained in the route list of RREP. If the cost is smaller than that of any previously received RREP, it verifies the packet, appends its own identifier to the route list, signs the packet, and broadcasts it. Finally, when *S* receives a reply, it can verify that it is from *D* and each hop in the route list is signed properly. Different from ARAN where only one possibly nonoptimal path is discovered, here *S* may receive multiple replies for different routes. Each route contains the full list of intermediate nodes and has a total cost. *S* can choose the one with minimum cost or smallest hop count for real data delivery.

Secure Packet Forwarding

The protection of routing message exchange is only part of the network-layer security solution for MANET. It is possible for a malicious node to correctly participate in the route discovery phase but fail to correctly forward data packets. The security solution should ensure that each node indeed forwards packets according to its routing table. This is typically achieved by the reactive approach because attacks on packet forwarding cannot be prevented: an attacker may simply drop all packets passing through it, even though the packets are carefully signed. At the heart of the reactive solutions are a detection technique and a reaction scheme, which are described as follows.

Detection — Because the wireless channel is open, each node can perform localized detection by overhearing ongoing transmissions and evaluating the behavior of its neighbors. However, its accuracy is limited by a number of factors such as channel error, interference, and mobility. A malicious node may also abuse the security solution and intentionally accuse legitimate nodes. In order to address such issues, the detection results at individual nodes can be integrated and refined in a distributed manner to achieve consensus among a group of nodes. An alternative detection approach relies on explicit acknowledgment from the destination and/or intermediate nodes to the source so that the source can figure out where the packet was dropped.

Localized detection. Reference [18] proposes *watchdog* to monitor packet forwarding on top of source routing protocols like DSR. It assumes symmetric bidirectional connectivity: if *A* can hear *B*, *B* can also hear *A*. Since the whole path is specified, when node *A* forwards a packet to the next hop *B*, it knows *B*'s next hop *C*. It then overhears the channel for *B*'s transmission to *C*. If it does not hear the transmission after a timeout, a failure tally associated with *B* is increased. If the tally exceeds a threshold bandwidth, *A* sends a report packet to the source notifying B's misbehavior.

Reference [19] follows the same concept but works with distance vector protocols such as ADOV. It adds a *next_hop* field in AODV packets so that a node can be aware of the correct next hop of its neighbors. It also considers more types of attacks, such as packet modification, packet duplication, and packet jamming DoS attacks. Each independent detection result is signed and flooded; multiple such results from different nodes can collectively revoke a malicious node of its certificate, thus excluding it from the network.

ACK-based detection. The fault detection mechanism proposed in [17] is based on explicit acknowledgments. The destination sends back ACKs to the source for each successfully received packet. The source can initiate a fault detection process on a suspicious path that has recently dropped more packets than an acceptable threshold. It performs a binary search between itself and the destination, and sends out data packets piggybacked with a list of intermediate nodes, also called *probes*, which should send back acknowledgments. The source shares a key with each probe, and the probe list is "onion" encrypted. Upon receiving the packet, each probe sends back an ACK, which is encrypted with the key shared with the source. The source in turn verifies the encrypted ACKs and attributes the fault to the node closest to the destination that sends back an ACK.

Reaction — Once a malicious node is detected, certain actions are triggered to protect the network from future attacks launched by this node. The reaction component typically is related to the prevention component in the overall security system. For example, the malicious node may have its certificate revoked, or be chosen with smaller probability in future forwarding paths. Based on their scope, the reaction schemes can be categorized as global reaction and end-host reaction. In the former scheme, all nodes in the network react to a malicious node as a whole. In

other words, the malicious node is excluded from the network. On the other hand, in the end-host reaction scheme, each node may make its own decision on how to react to a malicious node (e.g., putting this node in its own blacklist or adjusting the confidentiality weight of this node).

Global reaction. The reaction scheme in [19] falls into the global reaction category. It is based on the URSA certification framework [20]. Once multiple nodes in a local neighborhood have reached consensus that one of their neighbors is malicious, they collectively revoke the certificate of the malicious node. Consequently, the malicious node is isolated in the network as it cannot participate in the routing or packet forwarding operations in the future.

End-host reaction. The *pathrater* in [18] allows each node to maintain its own rating for every other node it knows about. A node slowly increases the rating of well-behaved nodes over time, but dramatically decreases the rating of a malicious node that is detected by its watchdog. Based on the rating, the source always selects the path with the highest average rating. Clearly each node may have a different opinion about whether another node is malicious, and each has its independent reaction accordingly. Reference [17] extends this idea with security protection of the routing messages, as discussed earlier.

Link-Layer Security

Link-layer security solutions protect the one-hop connectivity between two direct neighbors that are within the communication range of each other through secure MAC protocols. We use 802.11, the de facto standard MAC protocol for MANETs, to illustrate the link-layer security issues.

IEEE 802.11 MAC — The vulnerability of the IEEE 802.11 MAC to DoS attacks was recently identified. The attacker may exploit its binary exponential backoff scheme to launch DoS attacks [10, 11]. Reference [10] uses simulations to show that implementing a fair MAC protocol is a necessary but insufficient technique to solve the problem. A more robust MAC protocol with fairness guarantees is required to secure the MANET link-layer operations. Recently a security extension to 802.11 was also proposed in [11]. It follows the reactive approach and seeks to detect and handle such MAC-layer misbehaviors. The original 802.11 backoff scheme is slightly modified in that the backoff timer at the sender is provided by the receiver instead of setting an arbitrary timer value on its own. When a malicious node selects a small backoff value or does not back off at all, the receiver can detect such misbehaviors by checking the deviation between the actual transmission schedule and the expected schedule. The receiver then reacts by penalizing the misbehaving node and assigning larger backoff values to it.

The NAV field carried in the RTS/CTS frames exposes another vulnerability to DoS attacks [21]. Since the attacker in the local neighborhood is aware of the duration of the ongoing transmission, it may transmit a few bits within this period to incur bit errors in a victim's link-layer frame via wireless interference. Because the attacker can disrupt a legitimate frame of thousands or even tens of thousands of bits with little effort, the power consumption battle favors the adversary side rather than the legitimate node side. To the best of our knowledge, it remains unclear how to defeat such resource consumption DoS attacks in MANETs.

IEEE 802.11 WEP — It is well known that the IEEE 802.11WEP protocol [3] is vulnerable to attacks of two categories:

- Message privacy and message integrity attacks [2]. These attacks are based on various mechanisms such as short IV, linear cyclic redundancy check (CRC)-32 checksum, and key stream recovery by known plaintext attacks.
- Probabilistic cipher key recovery attacks such as the Fluhrer-Mantin-Shamir attack [22]. These attacks are based on the fact that the initial output in the RC4 key stream is disproportionally affected by a small number of key bits, particularly the prefix and postfix parts of the key [23].

Fortunately, the recently proposed 802.11i/WPA [24] has mended all obvious loopholes in WEP. Future countermeasures such as RSN/AES-CCMP [24] are also being developed to improve the strength of wireless security. We do not provide more details here because these cryptographic problems are not unique to ad hoc networks, and have been extensively studied in the context of wireless LANs.

Open Challenges

The research on MANET security is still in its early stage. The existing proposals are typically attack-oriented in that they first identify several security threats and then enhance the existing protocol or propose a new protocol to thwart such threats. Because the solutions are designed explicitly with certain attack models in mind, they work well in the presence of designated attacks but may collapse under unanticipated attacks. Therefore, a more ambitious goal for ad hoc network security is to develop a multifence security solution that is embedded into possibly every component in the network, resulting in in-depth protection that offers multiple lines of defense against many both known and unknown security threats. This new design perspective is what we call *resiliency-oriented* security design.

We envision the resiliency-oriented security solution as possessing several features. First, the solution seeks to attack a bigger problem space. It attempts not only to thwart malicious attacks, but also to cope with other network faults due to node misconfiguration, extreme network overload, or operational failures. In some sense, all such faults, whether incurred by attacks or misconfigurations, share some common symptoms from both the network and end-user perspectives, and should be handled by the system. Second, resiliency-oriented design takes a paradigm shift from conventional intrusion pre-

The research on MANET security is still in its early stage. The existing proposals are typically attack-oriented in that they first identify several security threats and then enhance the existing protocol or propose a new protocol to thwart such threats.

The solution relies on multiple fences, spanning different devices, different layers in the protocol stack, and different solution techniques, to guard the entire system. Each fence has all functional elements of prevention, detection/verification, and reaction.

vention to intrusion tolerance. In a sense, certain degrees of intrusions or compromised/captured nodes are the reality to face, not the problem to get rid of, in MANET security. The overall system has to be robust against the breakdown of any individual fence, and its performance does not critically depend on a single fence. Even though attackers intrude through an individual fence, the system still functions, but possibly with graceful performance degradation. Third, as far as the solution space is concerned, cryptography-based techniques just offer a subset of toolkits in a resiliency-oriented design. The solution also uses other noncrypto-based schemes to ensure resiliency. For example, it may piggyback more "protocol invariant" information in the protocol messages, so that all nodes participating in the message exchanges can verify such information. The system may also exploit the rich connectivity of the network topology to detect inconsistency of the protocol operations. In many cases, routing messages are typically propagated through multiple paths and redundant copies of such messages can be used by downstream nodes. Fourth, the solution should be able to handle unexpected faults to some extent. One possible approach worth exploring is to strengthen the correct operation mode of the network by enhancing more redundancy at the protocol and system levels. At each step of the protocol operation, the design makes sure what it has done is completely along the right track. Anything deviating from valid operations is treated with caution. Whenever an inconsistent operation is detected, the system can raise a suspicion flag and query the identified source for further verification. This way, the protocol tells right from wrong because it knows right with higher confidence, not necessarily knowing what is exactly wrong. The design strengthens the correct operations and may handle even unanticipated threats in runtime operations. Next, the solution may also take a collaborative security approach, which relies on multiple nodes in a MANET to provide any security primitives. Therefore, no single node is fully trusted. Instead, only a group of nodes will be trusted collectively. The group of nodes can be nodes in a local network neighborhood or all nodes along the forwarding path. Finally, the solution relies on multiple fences, spanning different devices, different layers in the protocol stack, and different solution techniques, to guard the entire system. Each fence has all functional elements of prevention, detection/verification, and reaction.

The above mentioned resiliency-oriented MANET security solution poses grand yet exciting research challenges. How to build an efficient fence that accommodates each device's resource constraint poses an interesting challenge. Device heterogeneity is one important concern that has been largely neglected in the current security design process. However, multifence security protection is deployed throughout the network, and each individual fence adopted by a single node may have different security strength due to its resource constraints. A node has to properly select security mechanisms that fit well into its own available resources, deployment cost, and other complexity concerns. The security solution should not stipulate the minimum requirement a component must have. Instead, it expects best effort from each component. The more powerful a component is, the higher degree of security or resiliency it has. Next, identifying the system principles of how to build such a new-generation of network protocols remains unexplored. The state-of-the-art network protocols are all designed for functionality only. The protocol specification fundamentally assumes a fully trusted and well-behaved network setting for all message exchanges and protocol operations. It does not anticipate any faulty signals or ill-behaved nodes. We need to identify new principles to build the next-generation network protocols that are resilient to faults. There only exist a few piecemeal individual efforts. Finally, evaluating the multifence security design also offers new research opportunities. The effectiveness of each fence and the minimal number of fences the system has to possess to ensure some degree of security assurances should be evaluated through a combination of analysis, simulations, and measurements in principle. However, it is recognized that the current evaluation for state-of-the-art wireless security solutions is quite ad hoc. The community still lacks effective analytical tools, particularly in a large-scale wireless network setting. The multidimensional trade-offs among security strength, communication overhead, computation complexity, energy consumption, and scalability still remain largely unexplored. Developing effective evaluation methodology and toolkits will probably need interdisciplinary efforts from research communities working in wireless networking, mobile systems, and cryptography.

REFERENCES

[1] C. Perkins and E Royer, "Ad Hoc On-Demand Distance Vector Routing," *2nd IEEE Wksp. Mobile Comp. Sys. and Apps.*, 1999.

[2] D. Johnson and D. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," *Mobile Computing*, T. Imielinski and H. Korth, Ed., Kluwer, 1996.

[3] IEEE Std. 802.11, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," 1997.

[4] B. Schneier, *Secret and Lies, Digital Security in a Networked World*, Wiley, 2000.

[5] Y. Hu, A. Perrig, and D. Johnson, "Ariadne: A Secure On-demand Routing Protocol for Ad Hoc Networks," *ACM MOBICOM*, 2002.

[6] M. Zapata, and N. Asokan, "Securing Ad Hoc Routing Protocols," *ACM WiSe*, 2002.

[7] B. Dahill *et al.*, "A Secure Protocol for Ad Hoc Networks," *IEEE ICNP*, 2002.

[8] Y. Hu, A. Perrig, and D. Johnson, "Packet Leashes: A Defense against Wormhole Attacks in Wireless Networks," *IEEE INFOCOM*, 2002.

[9] N. Borisov, I. Goldberg, and D. Wagner, "Intercepting Mobile Communications: The Insecurity of 802.11," *ACM MOBICOM*, 2001.

[10] V. Gupta, S. Krishnamurthy, and M. Faloutsos, "Denial of Service Attacks at the MAC Layer in Wireless Ad Hoc Networks," *IEEE MILCOM*, 2002.

[11] P. Kyasanur, and N. Vaidya, "Detection and Handling of MAC Layer Misbehavior in Wireless Networks," *DCC*, 2003.

[12] C. Perkins, and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," *ACM SIGCOMM*, 1994.

[13] P. Papadimitratos, and Z. Haas, "Secure Routing for Mobile Ad Hoc Networks," *CNDS*, 2002.

[14] A. Perrig *et al.*, "The TESLA Broadcast Authentication Protocol," *RSA CryptoBytes*, vol. 5, no. 2, 2002, pp. 2–13.

[15] Y. Hu, D. Johnson, and A. Perrig, "Sead: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks," *IEEE WMCSA*, 2002.
[16] P. Papadimitratos and Z. Haas, "Secure Link State Routing for Mobile Ad Hoc Networks," *IEEE Wksp. Security and Assurance in Ad Hoc Networks*, 2003.
[17] B. Awerbuch *et al.*, "An On-Demand Secure Routing Protocol Resilient to Byzantine Failures," *ACM WiSe*, 2002.
[18] S. Marti *et al.*, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," *ACM MOBICOM*, 2000.
[19] H. Yang, X. Meng, and S. Lu, "Self-Organized Network Layer Security in Mobile Ad Hoc Networks," *ACM WiSe*, 2002.
[20] J. Kong *et al.*, "Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks," *IEEE ICNP*, 2001.
[21] G. Noubir and G. Lin, "Low-Power DoS Attacks in Data Wireless LANs and Countermeasures," *ACM MobiHoc*, Poster Session, 2003.
[22] A. Stubblefield, J. Ioannidis, and A. Rubin, "Using the Fluhrer, Mantin, and Shamir Attack to Break WEP," *NDSS*, 2002.
[23] S. Fluhrer, I. Mantin, and A. Shamir, "Weakness in the Key Scheduling Algorithm of RC4," *8th Annual Wksp. Sel. Areas in Cryptography*, 2001.
[24] IEEE Std. 802.11i/D30, "Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Specification for Enhanced Security," 2002.

Additional Reading

[1] J. Hubaux, L. Buttyan, and S. Capkun, "The Quest for Security in Mobile Ad Hoc Networks," *ACM MobiHoc*, 2001.

Biographies

HAO YANG (hyang@cs.ucla.edu) received his B.S. degree from the University of Science and Technology of China in 1998 and his M.S. degree from the Chinese Academy of Sciences in 2001. Since then he has been pursuing his Ph.D. degree in the Computer Science Department at the University of California at Los Angeles (UCLA). His research interests include wireless networking, network security, and distributed systems.

HAIYUN LUO (hluo@cs.ucla.edu) received his B.S. degree from the University of Science and Technology of China in 1998 and his M.S. degree from UCLA in computer science in 2000. He is currently a Ph.D. candidate in the UCLA Computer Science Department. His research interests include wireless networking and mobile computing, security, and large-scale distributed systems.

FAN YE (yefan@cs.ucla.edu) is currently a Ph.D. candidate in the UCLA Computer Science Department. His research interests include wireless sensor networks, wireless network security, and computer networks.

SONGWU LU (slu@cs.ucla.edu) is currently an assistant professor in the UCLA Computer Science Department. He received both his M.S. and Ph.D. degrees from the University of Illinois at Urbana-Champaign. He received an NSF CAREER award in 2001. His research interests include wireless networking, mobile systems, sensor networks, and wireless network security.

LIXIA ZHANG (lixia@cs.ucla.edu) is currently a professor in the UCLA Computer Science Department. She received her Ph.D. degree from Massachusettes Institute of Technology. Her research interests include architecture and protocols for large-scale and high-speed networks, protocol design, implementation and analysis, and wireless sensor networks.

Developing effective evaluation methodology and toolkits will probably need interdisciplinary efforts from research communities working in wireless networking, mobile systems, and cryptography.