

Chapter 1

Authentication and Confidentiality in Wireless Ad Hoc Networks

Anjum Naveed and Salil Kanhere

*School of Computer Science and Engineering
University of New South Wales
Sydney, Australia, 2052
(anaveed,salilk)@cse.unsw.edu.au*

The security services of authentication and confidentiality are of significant importance to ensure secure communication in any network. The decentralized nature and the broadcast medium of communication of wireless ad hoc networks results in unique challenges in realizing the services of authentication and data confidentiality. In this chapter, we first highlight the issues relating to authentication and confidentiality in wireless ad hoc networks and identify the characteristics of these services. Subsequently, we discuss the security mechanisms proposed for authentication and confidentiality in wireless ad hoc networks. The chapter also includes a detailed discussion about the standards IEEE 802.1X (Authentication) and IEEE 802.11i (Confidentiality).

1. Introduction

A Wireless Ad Hoc Network is a group of low capacity computing devices (laptops, PDAs, etc.) connected through wireless links. These devices are generally mobile with frequent location changes. Communication between the devices can be established anywhere, in a decentralized manner without the support of an established infrastructure. The purpose of ad hoc networks is to enable the mobile device users to share resources, provide services to each other or simply establish a network for communication and information exchange. Ad hoc networks have a number of applications where infrastructure free communication is required. These applications include emergency relief, military operations, on-demand conferencing and home networking. Like any communication network, the true potential

of wireless ad hoc networks cannot be exploited without considering and adequately addressing the security issues.

ITU-T Recommendation X.800³ – Security Architecture for OSI – identifies the required security services for the communication networks. The security services have been broadly categorized into five groups namely authentication, access control or authorization, confidentiality, integrity and non-repudiation. Security management services that have been identified aim at ensuring availability, accountability and event management. Like all communication networks, wireless ad hoc networks require the same set of security services. However, the unique characteristics of ad hoc networks (decentralized communication, heterogeneous nature of devices, high mobility and frequently changing network topology) result in unique challenges to the security of ad hoc networks. The focus of this chapter is the security services of authentication and confidentiality with explicit consideration of the security challenges of wireless ad hoc networks.

The security service of authentication provides the assurance that any particular entity (wireless device) is the one who it claims to be. With the perspective of wireless ad hoc networks, the service of authentication is further divided into two components: (i) Access Authentication and (ii) Origin Authentication. The objective of access authentication is to ensure that only legitimate devices can access the network services. This in turn protects the network from illegal access and malicious jeopardization. On the other hand, the origin authentication ensures that within the authenticated network nodes, a node must be able to prove its identity for every communication session with any other node in the network. This ensures that an authenticated node cannot impersonate another legitimate node in the network. Consequently, the network is protected against misbehaving and compromised nodes.

One of the methods used in the Internet for authentication is asymmetric key cryptography. In this cryptographic technique the identity of the user/device is bound with a private and a public key. The public key is known to everyone while the private key is known only to the device that owns the key. Suppose device A intends to communicate with device B, it encrypts the message using its private key and a publically known encryption algorithm. Upon receiving the message, device B verifies if A transmitted the message by decrypting the message using public key of device A. If the message is successfully decrypted (correctness of a message is verified through Cyclic Redundancy Check, CRC), the message is considered to be originating from the authentic device A, otherwise, it is assumed that an unauthenticated device is impersonating the device A.

Confidentiality ensures that the information transmitted across the network is accessible only by the intended recipients. In the example of the preceding paragraph, to ensure the confidentiality of the information, device A encrypts the message using public key of device B. Upon receiving the message, device B decrypts the message using its private key. In this case, a device can decrypt the message successfully only if it is in possession of valid private key of device B. Since the private key of device B is only known to the device itself, only the device B can decrypt the message successfully, ensuring the message confidentiality. Authentication and confidentiality are discussed together in this chapter because mostly the same method and keying material is used to provide both services.

The rest of the chapter is organized as follows. In Section 2 we highlight the issues associated with authentication and confidentiality in ad hoc networks. Section 3 highlights the characteristics of these services for ad hoc networks. Section 4 and Section 5 detail numerous solutions proposed in literature for the services of authentication and confidentiality in ad hoc networks. We analyse the strengths, weaknesses and overheads of each solution. Section 6 is dedicated to the standardization effort for securing the communication in ad hoc networks. Section 7 enlists the open issues and Section 8 concludes the chapter.

2. Security Issues Relating Authentication and Confidentiality

In this section, we highlight the security issues relating authentication and confidentiality that arise from the unique characteristics of wireless ad hoc networks (limited resources, infrastructureless network and high mobility). We also discuss why the solutions proposed for wired networks and wireless local area networks (WLAN) are not feasible for wireless ad hoc networks.

Limited resources: The devices participating in a wireless ad hoc network are generally limited in computational and communicational resources. Most of the devices are battery operated with limited battery life. In wired networks, the information confidentiality can be achieved through strong asymmetric cryptographic solutions. However, limited computational resources and battery power render these solutions infeasible for wireless ad hoc networks. There is a tradeoff between the achievable level of security and the resources required to achieve the desired level. Higher level of confidentiality can only be achieved at the cost of scarce computational and battery resources.

Decentralized and infrastructureless network: The commonly used method of authentication in the Internet is through digital certificates. The certificates are issued by a centralized certification authority to the nodes within the network. A centrally located authentication server is responsible for validating the certificates presented by a node. The authentication server maintains a list of valid certificates. However, the decentralized nature of wireless ad hoc networks renders these methods impractical. Furthermore, computationally limited and highly mobile devices are incapable of acting as certification authority or the authentication server.

High mobility: One of the methods proposed for ad hoc networks to provide the services of certification authority and authentication is by replicating these services at different network nodes. Such a replication can reduce the overhead involved. However, only a limited number of highly trusted and secure nodes can be assigned the role of certification authority. High mobility of nodes may render such solutions impractical since the service providing nodes can become frequently unreachable by the service requesting nodes.

3. Characteristics of Security Services

Based on the issues highlighted in the previous section, the security solutions should possess the following characteristics in order to effectively provide the security services of authentication and confidentiality.

- The security services of authentication and confidentiality should induce minimum computational and communicational overhead.
- The level of security should be adjustable depending upon the level of resources (computational and battery power) available.
- The services should be scalable, considering large sized networks with frequent arrival and departure of the network nodes.
- The security services should not rely on any centralized entity, and no assumption should be made about the availability of any kind of infrastructure.
- Availability of authentication and confidentiality should be ensured, keeping in view the high mobility of the nodes.
- No assumption should be made about the node density while providing the security services.
- The security services should be robust against multiple malicious, compromised and misbehaving nodes since such scenarios are

frequent in an ad hoc networking environment. Moreover, the services should be resilient against attacks like identity theft, session hijacking, eavesdropping and sybil attacks.

4. Authentication in Wireless Ad Hoc Networks

The objective of authentication is to ensure that only legitimate devices can access the network services. The network nodes should be able to identify a malicious device, impersonating a legitimate participant node. Furthermore, if a participant device misbehaves after the trust relationship is established, the authentication mechanism should be able to evict the misbehaving node. Such a node should be denied of any further access to the network services or communication with any legitimate node in the network. A comprehensive analysis of the authentication protocols for wireless networks can be found in Ref. 16. The security techniques used to provide the authentication services can broadly be classified into three categories: (i) Symmetric cryptography, (ii) Asymmetric cryptography, and (iii) Collaborative mechanisms (i.e., Threshold cryptography). In this section, we explain the three categories in detail. We detail different solutions proposed for ad hoc networks based on these techniques with a discussion on the objectives of each solution, the employed approach, and strengths and weaknesses of the proposed solution. We also explain the important aspect of revocation, a mechanism used for evicting the misbehaving nodes or refreshing the authentication material for a compromised node.

4.1. *Symmetric Cryptographic Techniques*

The symmetric cryptographic techniques employ the use of a shared secret key among the participating nodes (pair of nodes intending to communicate or the nodes requesting access to the network) to provide the service of authentication. In its simplest form, a common key is issued to all legitimate nodes in the network. This key can be distributed manually to the participant nodes. Any node in possession of the key can authenticate itself by presenting the key and can access the network or any service offered by the network. The computational and communicational overheads involved in this kind of authentication are negligible. However, symmetric key based techniques are only suitable for small scale networks. The probability of the shared secret key being compromised increases proportionally with the increasing network size. Furthermore, if a single node is compromised,

the entire network is compromised. Therefore, the secret key needs to be changed frequently in order to ensure the appropriate level of security.

Wired Equivalent Privacy (WEP) protocol: the security mechanism initially employed by the IEEE 802.11i standard for WLAN security⁴ is predominantly based on the symmetric cryptographic technique. In addition to the issues identified above, a number of additional security issues have been identified. We do not go into details of these issues here. Interested readers are referred to the related publications.^{20–22}

4.2. *Asymmetric Cryptographic Techniques*

Asymmetric cryptographic solutions involve the use of a pair of keys (a public key and a private key) for each participating node. The private key is known only to the node to whom it was issued, while the public key of the node is known to all the participating nodes. These keys are pre-distributed (often before joining the network) to the nodes by a Certification Authority in form of a digital certificate. A digital certificate binds the node identity with the two keys and associates an expiration time with the certificate. The certificate is then signed by the private key of the certification authority to make it tamper proof. To authenticate itself and to access the network services (join the network or start a communication with a member node) a node presents its digital certificate. The existing member nodes can extract the information stored in the certificate by decrypting the certificate using the public key of the certification authority, which is distributed among all participant nodes. The validity of the certificate can then be verified to ensure that the certificate was issued to the node which is presenting the certificate and that the certificate has not expired yet. If the certificate is valid, the node is allowed to access the services it requested, otherwise, the node is considered a malicious attacker and is denied network access. Note that the certificate of the misbehaving nodes can be revoked. This is achieved through a certificate revocation list that can be maintained at a centralized location where all the revoked certificates can be listed.

Several issues specific to ad hoc networks are involved with the above mentioned authentication technique: (i) the decentralized and infrastructure free nature of ad hoc networks make it impractical to have a centralized certification authority; (ii) computational and communicational overhead can be significantly high in case of asymmetric cryptography; and (iii) in the case of misbehaving nodes, certificate revocation can be a challenging task as a centralized certificate revocation list maintenance is impractical.

Nevertheless, the strengths of asymmetric cryptographic techniques have encouraged the researchers to employ the technique for authentication and security of wireless ad hoc networks. Several solutions to the above mentioned problems have been proposed in the literature. In Section 4.3, we discuss the techniques that focus on the distribution of the certification authority and authentication server within the network nodes. Section 4.4 details the key revocation mechanisms employed in wireless ad hoc networks.

4.3. Collaborative Mechanisms

In this section, we present authentication mechanisms that aim at distributing the role of the certification authority among the participating nodes.

The idea of utilizing threshold cryptography for collaborative authentication in ad hoc networks was first proposed by Zhou *et al.*¹¹ Since then, a number of distributed neighbor collaboration authentication protocols have been proposed by researchers following a similar approach.^{12–14,17} For example, Deng *et al.*¹⁷ have proposed threshold cryptography based solution for the distribution of the master key \langle public key, private key \rangle . A node in the network is authenticated through its private key. In the proposed scheme, all nodes possess the public key while every node has a share of the private key. The (k,n) threshold secret sharing scheme is employed to generate the private key for a node which states that k out of n shares of private key are required to construct the complete private key and less than k shares of the secret key cannot construct the complete private key. Based on this mechanism, whenever a node needs to refresh its private key, it needs k neighbors to send their secret share to the node to reconstruct the private key, and no node can construct the private key based on its own information. The process of private key generation is shown in the Figure 1 where the requesting node broadcasts the *REQUEST* message along with its own share for verification. The neighboring nodes reply to the *REQUEST* message by sending their own share of the secret key to the requesting node. The requesting node is then able to generate the private key on receiving k shares of the key. In this way, an intruding node cannot generate the private key unless its own share of private key is verified by k neighboring nodes. Similarly, the private key of a misbehaving node is not refreshed by the neighbors. Therefore, the threshold secret sharing serves as the strong authentication and key management solution.

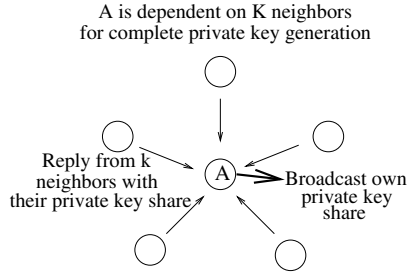


Fig. 1. Neighbor collaboration for private key generation in Wireless Mesh Networks.

Note that threshold cryptography is based on the asymmetric cryptography where the private key is distributed among nodes as partial components. Consequently, the computational overheads involved in the schemes detailed above can be significantly higher. The process of combining the partial shares to generate the private key can incur additional delay. Furthermore, if a particular node does not have enough number of neighbors to collect k shares of the private key, its key will be revoked even if the node is legitimate and a well behaving member.

Capkun *et al.*²⁹ have proposed a self-organized public-key management scheme for mobile ad hoc networks. The authors propose that each node issues its own public/private key pair, which binds the node identity with the issued keys. The neighboring nodes of a particular node can verify if a particular public key is associated with the identity of the node that claims the key. Based on this verification a certificate can be issued (signed using private key of the issuing node) and distributed to the nodes within the network. The certificates are issued for a specific time interval and renewed after that interval of time, provided that the identity of the node and the key binding is still trusted. This mechanism works as a distributed certification authority. With random mobility of the nodes and the exchange of certificate repositories, the local certificate repositories of the nodes can quickly grow, establishing the trust relationship with other nodes. The certificates are revoked explicitly if misbehavior or invalid identity-key binding is detected based on inconsistency between the neighboring node repositories. The key authentication of a node v by a node u is performed through chain of certificates as follows: (i) The first certificate of the chain should be directly verifiable by u . i.e., signed by the public key of u . (ii) Each remaining certificate can be verified using public key contained in the previous certificate of the chain. (iii) The last certificate contains the public key

of v . The proposed mechanism works without requiring a centralized certification authority or authentication server. However, the proposed system can be throttled by multiple colluding adversaries.

Keoh *et al.*¹³ have proposed similar credential verification scheme that establishes a web of trust between the nodes. However, the authors suggest the association of different trust levels for each credential verification by the nodes. Furthermore, partial attributes of the XML based Credential Assertion Statements can also be verified by the nodes. Interested readers are referred to the related publication for further details.¹³

4.4. Certificate Revocation for Wireless Ad Hoc Networks

Wireless ad hoc networks are prone to security attacks – where one or few participating nodes are compromised by external adversary to gain network access – due to the broadcast nature of their communication. Furthermore, the cooperative nature of these networks leads to several attacks launched by internal selfish nodes. These misbehaving nodes (compromised or selfish) need to be identified and evicted from the network to ensure secure and smooth operation. The authentication credentials of such nodes should be invalidated and all participating nodes should be informed about this invalidation, ensuring the isolation of the misbehaving nodes from the network. In wired networks, Certificate Revocation Lists (CRL) are maintained at some centralized, publically accessible location.²³ This list comprises of the information about the revoked certificates. Network nodes can access the list from a centralized location. Alternatively, the list can be broadcasted at regular intervals to the participating nodes.

In wireless ad hoc networks, the assumption of a centralized location is impractical and a single computationally limited node cannot be responsible for maintaining and broadcasting CRL. Furthermore, keeping in view the dynamic nature of wireless ad hoc networks, identification of misbehaving nodes can be a challenging task. This is due to the lack of uniform traffic patterns where anomalies can be detected. Several techniques have been proposed in the literature for identification of misbehaving nodes.^{2,24} However, misbehavior detection is not the focus of this chapter. We assume that one of these techniques^{2,24} can be employed to detect the misbehaving nodes. The techniques discussed in the following paragraphs aim at isolating the misbehaving nodes from the network by invalidating their authentication credentials.

Claude *et al.*²⁵ have proposed a certificate revocation scheme for wireless ad hoc networks. The authors propose that each node maintains a CRL

and the certificates are revoked by individual nodes. Each node maintains a profile table and a status table. When a node detects the misbehavior of a neighboring node, it launches an accusation against the misbehaving node by broadcasting the certificate identity of the misbehaving node. When a node receives the accusation message, it records the accusation in the profile table with the certificate identity of the accuser, the accused, the status of the certificate as broadcasted by the accuser and date and time of accusation. The message with duplicate accuser and accused certificate ID are ignored. The Status table is constructed from the profile table. It contains the following information: number of accusations against node i , number of accusations by node i , behavior index of node i , weight of i 's accusation, revocation quotient and certificate status of node i . The behavior index of node i is inversely proportional to the number of accusations against it. The weight of its accusation is inversely proportional to its behavior index and the number of accusations it makes against other nodes. The revocation threshold is used to decide if the certificate of the node should be revoked. When a node revokes the certificate of another node in the network, it no longer considers this node as authentic. Its accusations about other nodes are ignored and no communication from this node is trusted. However, the decision of the accusing node does not impact the decision of the neighboring nodes. The proposed method, while simplistic and efficient, involves excessive communication and storage overhead on the nodes.

Yang *et al.*² have proposed a distributed certificate renewal and revocation technique based on the threshold secret-shared cryptographic technique. The authors have utilized the routing behavior of the nodes to identify the misbehaving nodes. A novel token-based crediting scheme has been proposed. Each participant node is issued a token with a fixed token life when it joins the network. The token serves the purpose of a credential for node authentication. The token of the node expires after a fixed time duration. Just before token expiration, the node requests its neighboring nodes for token renewal. Note that the nodes maintain a credit index of each neighboring node within the network. k out of n neighbors collaborate using threshold secret-shared cryptography (discussed in previous section) to issue a new token to the requesting node. The token expiry time of this newly issued token depends upon the credit of the node. The credit of well behaving nodes gets accumulated over the period of time. Therefore, the token expiry time of these nodes is longer and is linearly incremented every time the node refreshes its token. On the other hand, the token of misbehaving nodes is revoked because enough number of neighbors (k) are

not willing to renew the token due to the node's bad credit. Furthermore, a node with an expired token is unable to participate in the network.

5. Confidentiality in Wireless Ad Hoc Networks

We will now analyze different privacy mechanisms proposed for ad hoc networks. For each solution we will address the approach, security issues addressed by the solution, strength of solution, overheads caused and the weaknesses.

Wu *et al.*²⁶ have proposed that the traffic from a source to destination can be split into multiple flows, each following a different routing path. Furthermore, if the traffic is split in a random way, the traffic pattern can be completely concealed from any intermediate adversary. Consequently, traffic confidentiality can be ensured in a multi-hop wireless network. The proposed technique, however, is dependent upon the use of a multi-path routing protocol that can find multiple diverse paths from every source to destination. Moreover, considerable overhead is involved at the destination where packet reassembly from different flows is required. Another drawback of the proposed scheme is that certain component flows will have to use sub-optimal and low quality paths, resulting in increased end-to-end delay.

Bouam *et al.*²⁷ have proposed a similar solution that ensures the data confidentiality. The authors have proposed splitting each message (instead of flows) into $n - 1$ components ($n \geq 3$), each component being routed on a separate path. A random number x is generated between 1 and n . Each message is associated with a unique identifier to help the receiver reassemble the message in order. The x^{th} component of the message is encrypted using symmetric encryption and transmitted to the receiver. The remaining $n - 1$ components are transmitted in pairs using XOR operation with reference to x . The operation is shown in Figure 2 for $x = 3$ and $n = 6$. The values of x and n are transmitted on a pre-decided control channel. The transmitted messages are also encrypted using symmetric encryption, resulting in a second level of security. The receiver first receives the x^{th} component and using this component, regenerates the remaining message components by repeatedly applying the XOR operation. The components are then placed in order to regenerate the original message. The proposed technique is flexible because Diversity Coding (which splits a single communicational channel into multiple component channels) can be used instead of multi-path routing. Furthermore, asymmetric cryptography can be used to encrypt each component message to provide stronger level of confidentiality

and origin authentication. However, the message splitting and reassembling can introduce considerable computational overhead and additional delay.

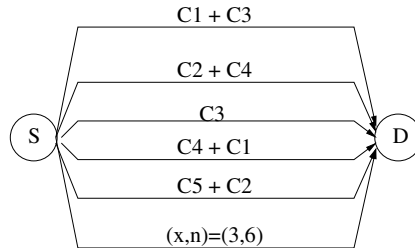


Fig. 2. Message splitting for data confidentiality.

Another technique proposed by Michell *et al.*²⁸ is a state based key hop protocol based on stream ciphers. The authors have proposed the use of the RC4 algorithm, however, unlike WEP where the state of RC4 is reinitialized for every packet, the authors propose that same seed should be used to generate the streams for a specific duration of time. The issue of a weak key is avoided by using a stream offset. The offset indicates the starting point down the stream from which the packet encryption should start. The two communicating nodes are synchronized such that the nodes know the initial seed for the stream (Base key), duration for which the key remains valid, RC4 states, and the offset. The RC4 states define the offset for the subsequent packets after the first packet is transmitted using initial offset. After the fixed duration, the base key expires and new base keys should be achieved (hence the name key hop) and the states should be re-synchronized by the two nodes. Note that no method is detailed for distribution of the synchronization parameters among the two nodes. The proposed algorithm offers a strong and light weight encryption algorithm and reduced computational overhead. However, the issue of distributing the parameters required for synchronization is not addressed. Knowledge of these parameters can enable an adversary to easily decrypt the entire communication between any pair of nodes. Therefore, a secure method for distribution of these parameters is of utmost importance for the secure operation of the protocol.

Soliman and Omari⁵ have proposed a security framework based on stream cipher for encryption to provide the services of data confidentiality,

data integrity, and authentication. This framework ensures per packet mutual authentication between the two communicating nodes within the network. The objective of using stream cipher is to allow online processing of the data. Consequently, minimum delay is introduced because of the security provisioning. Two secret security keys, Secret Authentication Key (SAK) and Secret Session Key (SSK), are used for authentication of the supplicant and authenticator. SAK is exchanged between the supplicant and the authenticator after initial mutual authentication from the authentication server, whereas the SSK is used for a given communication session between the two nodes. The SAK and SSK pair is used by the communicating nodes to generate the permutation vector (PV) which is used for the encryption and decryption of data. In the strongest mode of security, the data is also involved in the PV generation, resulting in the randomness which makes the decryption of the data difficult even if the encryption key of one packet is compromised. The synchronization of the generated permutation vector between the sender and the receiver of the data results in origin authentication of every MPDU. To minimize the security overhead, plain text MPDU is XORed with the PV generated for that MPDU. The authors have proved that the encryption of data using PV provides strong security services of data confidentiality, data integrity, and origin authentication.

Junaid *et al.*¹⁵ have proposed a piggyback challenge-response protocol, which relies on Advanced Encryption Standard (AES) in Counter Mode³⁰ for providing data confidentiality. AES in counter mode requires a counter block and an encryption key to encrypt the message. The message is divided into blocks of 128 bits and each block is encrypted using the encryption key and a unique counter block (see Ref. 30 for details). The authors propose the extension to IEEE 802.11i⁴ key generation mechanism as shown in Figure 3. The temporal key (TK) generated through IEEE 802.11i using four-way handshake is used as seed for the pseudo-random function (PRF-128) to generate the initial counter. This initial counter is used as the AES initial counter block, which is linearly incremented to generate subsequent counter blocks. The initial counter is also used as the first nonce N_0 , which is transmitted with the first message.

Data confidentiality and origin authentication is provided as follows: Suppose node A and node B share a PMK and wish to communicate. Assume that node A initiates the communication by sending an initial message to node B. Node A will use TK as the encryption key for this message. It

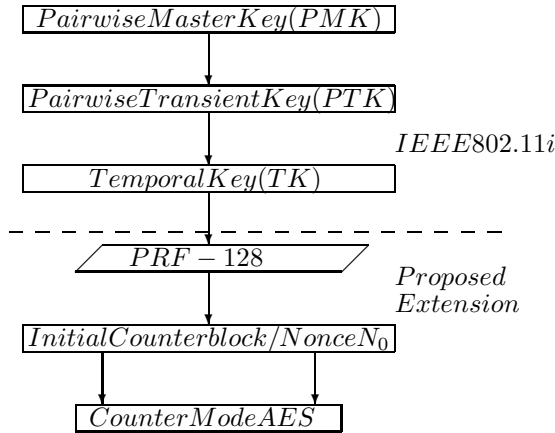


Fig. 3. Key generation mechanism.

will encrypt the first message along with the nonce N_0 (generated using Fig. 3) and the *Meta Data* using Eq. (1).

$$\mathbf{E}_{TK}(N_0||Data||MetaData) \quad (1)$$

The field *Meta Data* is used for message integrity and is beyond the scope of this chapter. The intended recipient (node B), upon receiving the message will also generate the initial counter (also the nonce N_0) using the procedure shown in Fig. 3. It will decrypt the message using Eq. (2), TK being the decryption key. After decryption, node B will compare its own generated nonce value with the received nonce. Since both nodes A and node B share the PMK, the N_0 generated should be the same as the N_0 which was transmitted as a part of the message by node A. The nonce will act as challenge text to authenticate the source of the message.

$$\begin{aligned} \mathbf{D}_{TK}(\mathbf{E}_{TK}(N_0||Data||MetaData)) \\ = N_0||Data||Metadadata \end{aligned} \quad (2)$$

Node B will then use N_0 as the encryption key for the reply, rather than the TK. PRF-128 will be used to generate a new nonce N_1 , which will be concatenated with the data and *Meta Data*, encrypted using N_0 and transmitted back to Node A. Thus, a new nonce is generated iteratively for each subsequent message, which enhances the robustness of the security

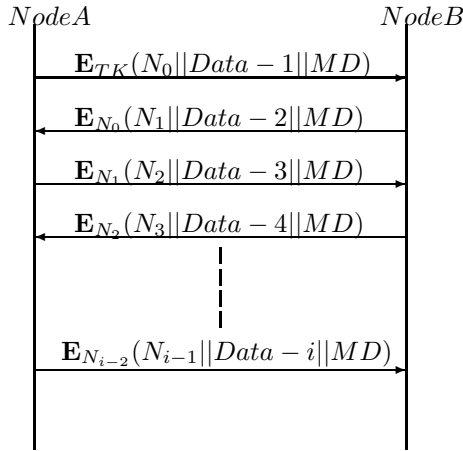


Fig. 4. Per frame authentication mechanism.

solution. Node A will employ the aforementioned decryption process to retrieve the message and authenticate (using the response nonce) the sender. The communication between nodes A and B is shown in Fig. 4. In general, the i -th message exchanged between nodes A and node B is encrypted using Eq. (3) and the corresponding decryption process uses Eq. (4). The exchange of the nonce results in a continuous challenge-response protocol, which provides data confidentiality as well as per packet authentication. The per packet authentication protects against MAC spoofing attacks as well as replay attacks.

$$\mathbf{E}_{N_{i-2}}(N_{i-1} || Data || MetaData) \tag{3}$$

$$\begin{aligned} \mathbf{D}_{N_{i-2}}(\mathbf{E}_{N_{i-2}}(N_{i-1} || Data || MetaData)) \\ = N_{i-1} || Data || Metadata \end{aligned} \tag{4}$$

6. Standardization Efforts

IEEE 802.11i⁴ is the defined standard for the MAC layer security of the wireless networks. We dedicate this section to discuss the IEEE 802.11i standard. The section begins with the explanation of the security methods used for the services of authentication and confidentiality in the IEEE 802.11i standard. Subsequently, we expose the vulnerabilities in IEEE

802.11i that render the standard prone to security attacks. These weaknesses lead to attacks including: pre-computation and partial matching attacks; session hijacking attacks; man-in-the-middle attacks exploiting vulnerabilities in IEEE 802.1X; and DoS attack exploiting vulnerabilities in four-way handshake. We also briefly discuss the proposed prevention mechanisms for these attacks.

IEEE 802.11i standard consists of three components: Key Distribution component, Mutual Authentication component, Data Confidentiality, Integrity, and Origin Authentication component. In the following sections, we briefly discuss these components under broad categories of authentication and confidentiality.

6.1. Mutual Authentication Using IEEE 802.1X Standard

IEEE 802.11i standard uses IEEE 802.1X⁶ for *key distribution and authentication*. IEEE 803.1X relies on Extensible Authentication Protocol (EAP)⁷ and an authentication, authorization, and accounting server (AAA Server) like RADIUS or DIAMETER^{8,9} for the purpose. IEEE 802.1X is a port-based access control protocol, which operates in a client-server architecture. When an authenticator (A member node of the network) detects a new supplicant (a node requesting to join the network), the port on the authenticator is enabled and set to the “unauthorized” state for that supplicant. Only 802.1X traffic (EAP messages) is allowed in this state. Any other traffic originating from the supplicant is blocked until after authentication. The authenticator sends out the EAP-Request message to the supplicant, which is replied by an EAP-Response message, containing the preloaded credentials of the supplicant. The authenticator forwards this message to the AAA server. The server may be distributed or replicated on several nodes in the case of wireless ad hoc networks. If the server authenticates the supplicant and accepts the request, it generates Pairwise Master Key (PMK), which is distributed to authenticator and supplicant using EAP messages. After authentication from server, the authenticator sets the port for the supplicant to the “authorized” state and normal traffic is allowed.

After successful distribution of the encryption key (PMK) and authentication of supplicant using 802.1X, the supplicant (mobile device) and the authenticator (peer mobile device) *mutually authenticate* each other. This process is based on the four-way handshake. The four-way handshake is initiated when the two nodes intend to exchange data. Although an encryption

key PMK is available to both the supplicant and the authenticator, this key is meant to last the entire session and should be exposed as little as possible. The purpose of four-way handshake is to use the PMK and establish two more keys called the Pairwise Transient Key (PTK) and Group Temporal Key (GTK).

The first message of the four-way handshake is transmitted by the authenticator to the supplicant which consists of ANonce. The supplicant uses this ANonce and readily available fields: Supplicant nonce (SNonce); Authenticator MAC address; and Supplicant MAC address, to generate the PTK using cryptographic hash function. The second message of the handshake is transmitted by the supplicant to the authenticator consisting of SNonce and Message Integrity Code (MIC), which is encrypted using PTK. The authenticator is then able to generate the PTK and GTK. The attached MIC is decrypted using the generated PTK. If the MIC is successfully decrypted, then the authenticator and the supplicant have successfully authenticated each other (Mutual Authentication). This is because the authenticator's generated PTK will only match the PTK transmitted by the supplicant if the two share the same PMK. Third message is transmitted by the authenticator consisting of GTK and MIC. The Last message of four-way handshake is the acknowledgement transmitted by the supplicant. The two nodes can exchange the data after successful four-way handshake.

PTK is used to generate Temporal Key (TK), which is used to encrypt unicast messages, while the GTK is used to encrypt broadcast and multicast messages. The four-way handshake (shown in Figure 5) involves generation and distribution of these keys between supplicant and authenticator and also leads to the mutual authentication of the two.

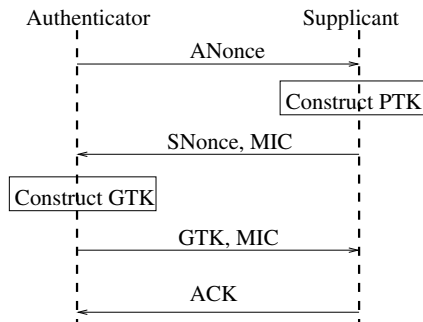


Fig. 5. Four-way handshake.

6.2. Confidentiality and Origin Authentication

IEEE 802.11i provides two methods for the security services of *data confidentiality and origin authentication*. First method, Temporal Key Integrity Protocol (TKIP) is the enhanced version of Wired Equivalent Privacy (WEP) and has been provided for backward compatibility with the hardware that was designed to use WEP. RC4 encryption is used in TKIP. We do not discuss TKIP here, interested readers are referred to the Section 8.3.2 of the standard⁴ for further details.

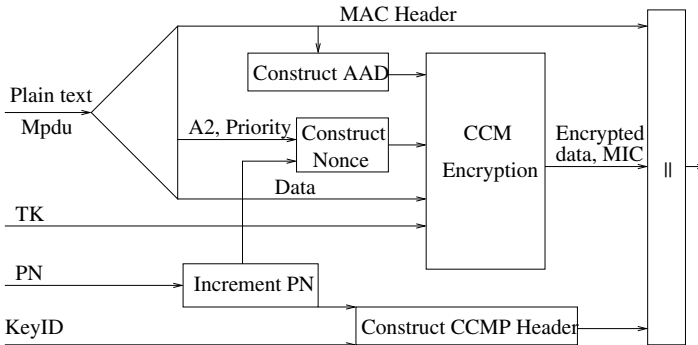


Fig. 6. CCMP encryption process and encrypted frame generation.

The second method is the Counter mode (CTR) with CBC-MAC Protocol (CCMP). CCMP is based on the Counter mode With CBC-MAC (CCM)¹⁰ of the AES encryption algorithm. CCM combines Counter (CTR) for confidentiality and the Cipher Block Chaining (CBC) Message Authentication Code (MAC) for origin authentication. CCM encryption takes four inputs, as shown in Figure 6: the Encryption key, Additional Authentication Data (AAD), a unique Nonce for every frame and the Plain text. CCM requires a fresh temporal key (TK) for encryption in every session. AAD is constructed from the MAC header and consists of the following fields: Frame Control field FC (Certain bits masked), Address A1, Address A2, Address A3, Sequence Control field SC (Certain bits masked), Address A4 (If present in the MAC header) and Quality of service Control field QC (if present). CCMP uses the A2 and the priority fields of MAC header along with a 48-bit Packet Number (PN) to generate the unique nonce value for each frame protected by a given TK. PN is incremented for each MAC Protocol Data Unit (MPDU) resulting in a fresh value of nonce for each

MPDU. The output of the encryption is the cipher text and the Message Integrity Code (MIC). The frame to be transmitted is constructed by concatenating the MPDU header, CCMP header, cipher text and MIC. CCM encryption is explained in RFC 3610.

6.3. Vulnerabilities in IEEE 802.11i and Security Attacks

A number of security vulnerabilities have been identified in the IEEE 802.11i standard. This section details these vulnerabilities, the attacks launched by exploiting the vulnerabilities and the available prevention mechanisms.

6.3.1. IEEE 802.1X Vulnerabilities

IEEE 802.1X⁶ is used for key distribution and authentication in IEEE 802.11i. The process of authentication involves three entities: Authenticator, Authentication Server and the Supplicant. The protocol assumes that the authenticator is always trusted. Therefore, the supplicant does not verify the messages received from the authenticator and unconditionally responds to these messages. This assumption is the security vulnerability that can be exploited by the adversary. The adversary can act as authenticator and launch the session hijacking attack and the man-in-the-middle attack as exposed in Ref. 19. Figure 7 shows how an adversary can launch session hijacking attack by exploiting the explained vulnerability. The adversary waits until the authenticator and the supplicant complete the authentication process and the authenticator sends the EAP success message to the supplicant. Following this, the adversary sends 802.11 disassociate message to the supplicant with the spoofed IP of the authenticator. The supplicant assumes its session has been terminated by the authenticator as the message is not verified for integrity. There onwards, the adversary gains the access to the network by spoofing the MAC address of supplicant and proceeds with mutual authentication procedure using four-way handshake.

Figure 8 shows man-in-the-middle attack launched by the adversary exploiting the same vulnerability. After the initial exchange of EAP request and response messages between the supplicant and the authenticator, the adversary sends EAP success message to the supplicant using its own MAC address. Since the IEEE 802.1X protocol suggests unconditional transition upon receiving the EAP success message by the supplicant, the supplicant assumes it is authenticated by the authenticator and changes the state. When the authenticator sends the EAP success message, the supplicant

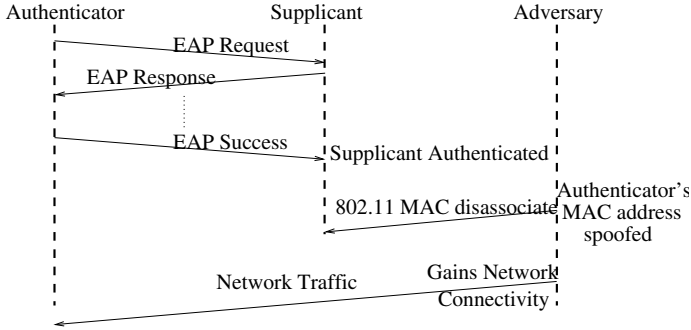


Fig. 7. Session hijacking attack on 802.1X authentication mechanism.

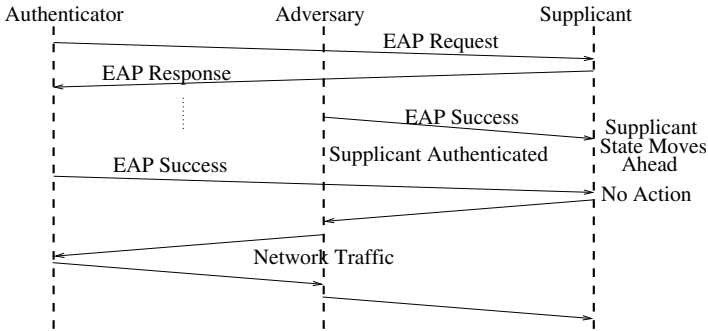
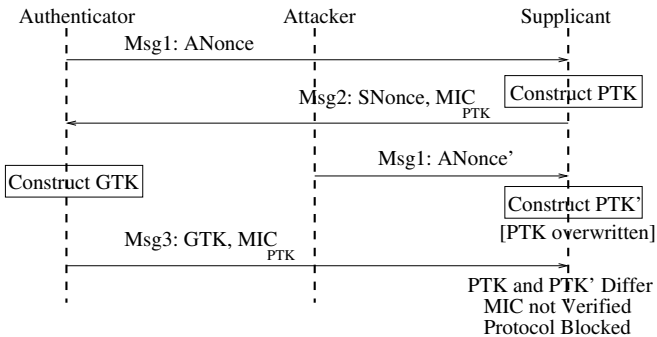


Fig. 8. Man-in-the-middle attack on 802.1X authentication mechanism.

has already passed the stage where it was waiting for the success message and hence no action is taken for this message. The supplicant assumes the adversary as the legitimate authenticator while the adversary can easily spoof the MAC address of the supplicant to communicate with the actual authenticator. Therefore, the adversary will become the intermediary between the supplicant and the authenticator. The proposed solutions to prevent these attacks¹⁹ recommend the authentication and integrity check for the EAP messages between the authenticator and the supplicant. The solution also proposes that the peer-to-peer based authentication model be adopted where the authenticator and the supplicant should be treated as peers and the supplicant should verify the messages from the authenticator during the process of trust establishment. The peer-to-peer model is suitable for wireless ad hoc networks where both the authenticator and the supplicant are wireless peer devices.

6.3.2. Four-way Handshake Vulnerabilities

Four-way handshake is the mechanism used for the mutual authentication of the supplicant and the authenticator in IEEE 802.11i. Vulnerabilities in the four-way handshake have been identified and the DoS attack exploiting these vulnerabilities proposed in Ref. 18. The handshake starts after PMK is distributed to the supplicant and the authenticator. The supplicant waits for a specific interval of time for message 1 of the handshake from the authenticator. If the message is not received, the supplicant disassociates itself from the authenticator. Note that this is the only timer used by the supplicant. If message 1 is received by the supplicant, it is then bound to respond to every message from the authenticator and wait for the response until it is received. On the other hand, the authenticator will timeout for every message, if it does not receive the expected response within a specific time interval. Further, the supplicant is de-authenticated if the response is not received after several retries. Also note that both the authenticator and the supplicant drop the message silently, if the MIC of the message cannot be verified.



(Attacker sends messages with spoofed MAC address of Authenticator)

Fig. 9. Denial-of-Service attack on four-way handshake.

This mechanism of four-way handshake is vulnerable to the DoS attack by the adversary. Consider Figure 9 where the authenticator sends the message 1 to the supplicant. Note that message 1 is not encrypted. Supplicant generates a new SNonce and then generates PTK using the ANonce, SNonce and other relevant fields and responds with the message 2, which is encrypted using PTK. At this point, the adversary sends the malicious

message 1 with the spoofed MAC address of the authenticator. The supplicant is bound to respond to the message. It assumes that the message 2 that it sent to the authenticator is lost so the authenticator has retransmitted the message 1. Therefore, it calculates PTK' (different from PTK and over writing PTK) based on the $ANonce$ sent by adversary and sends message 2 again which is encrypted using PTK' . Meanwhile, the authenticator responds to the first message 2 of the supplicant by sending the message 3 that is encrypted using PTK . The integrity check performed by the supplicant on message 3 fails, because the supplicant is now using PTK' while the authenticator encrypted the message using PTK . Consequently the four-way handshake process is blocked until the authenticator de-authenticates the supplicant after several retries, denying the supplicant of the service.

Three solutions have been proposed in Ref. 18 to prevent the above attack. We only discuss the most effective solution here. Note that every time the supplicant receives message 1, it generates a new $SNonce$ which is concatenated with $ANonce$ (transmitted by authenticator in message 1) and other relevant information to generate new PTK . The proposed solution suggests that the supplicant should store the $SNonce$ created in response to the first message 1 that it receives from authenticator. The same $SNonce$ should be used for all subsequent message 1s until the supplicant receives message 3 from the authenticator. Upon receiving the message 3, supplicant should use the newly transmitted $ANonce$ in message 3 and the stored $SNonce$ to generate PTK again, which should be used to decrypt message 3. Use of same $SNonce$ and $ANonce$ will generate same PTK if other information remains unchanged. Therefore, supplicant will be able to respond to the legitimate message 3 even if it receives multiple message 1s from adversary. Note that the adversary cannot send a malicious message 3 because message 3 is encrypted using PTK , which is dependent on PMK (only known to the supplicant and the authenticator).

6.3.3. *CCMP Encryption Vulnerabilities*

Although CCMP (employed by IEEE 802.11i) uses the CCM encryption, the strength of which is time tested, the protocol is vulnerable to the partial matching and pre-computation attacks. The vulnerabilities of the protocol implementation and the resulting attacks have been exposed in Ref. 15. The research shows that the address field $A2$ and the priority field of the MAC header and the PN field in the CCMP header are transmitted as plain text in the headers as well as in the encrypted form as part of the MIC. This

leads to the partial matching attack and the researchers have shown that the key strength of the 128-bit encryption key used in CCMP decreases. The decrease in the key strength exposes the protocol to pre-computation attack, resulting in the compromise of data confidentiality and data integrity. Further, The CCM encryption is a two phase process. During first phase the MIC is calculated and in the second phase the encryption of the frame takes place. Similarly, the decryption is done in two phases, where first the message integrity is verified from MIC and then the decryption takes place. The two phase processing of the frame at each wireless link may lead to considerable delay in the case of multi-hop wireless networks like wireless ad hoc networks where the data traverses a number of intermediate wireless hops before reaching the wired Internet. The delay introduced by the security services leads to the impracticability of the CCMP protocol for large wireless mesh networks consisting of several intermediate hops.

7. Open Issues

The strong security offered by asymmetric cryptography makes it an attractive solution for wireless ad hoc networks. However, the limited computational and communication capabilities of the devices and the unavailability of centralized certification and authentication servers pose challenges for adopting the asymmetric cryptographic solutions for wireless ad hoc networks. A number of light weight security solutions have been proposed as an alternative to asymmetric cryptography, reducing the complexity of security provisioning. However, majority of these protocols do not specify any mechanism for initial credential distribution and verification. Shared secret threshold cryptographic solutions can effectively address the problem of unavailability of centralized certification and authentication server. However, the additional overheads involved in key renewal and partial key accumulation to generate the private key make these solutions less attractive. Consequently, the wireless ad hoc networks have yet to meet a complete security solution for authentication and data confidentiality that is lightweight, distributed and covers all aspects of the two security services including initial credential distribution and verification.

8. Conclusion

In this chapter, we considered the two security services of authentication and data confidentiality in wireless ad hoc networks. The security issues

relating to authentication and confidentiality, specific to ad hoc networks, have been identified and the characteristics of these services have been outlined. The proposed security solutions for the two services of authentication and confidentiality have been categorized into three categories, depending upon the underlying security techniques. The proposed solutions within each category are discussed in detail. Finally, IEEE 802.11i, standard for wireless security is detailed, its vulnerabilities are highlighted and the solutions proposed for the vulnerabilities are discussed. The chapter ends with a note on the open issues relating the two security issues of authentication and confidentiality in wireless ad hoc networks.

References

1. Arunesh Mishra, William A. Arbaugh, *An Initial Security Analysis of the IEEE 802.1X Standard*, Technical report, university of Marryland. February 2002.
2. Hao Yang, Shu. J, Xiaoqiao Meng, Songwu Lu. *SCAN: self-organized network-layer security in mobile ad hoc networks*, Appears in: IEEE Journal on Selected Areas in Communications, Volume: 24, Issue: 2, pages 261-273, Februry 2006.
3. Security Architecture for Open Systems Interconnection for CCITT Applications, *ITU-T Recommendation X.800*, March 1991.
4. IEEE Std. 802.11i-2004, Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Medium Access Control (MAC) Security Enhancements. <http://standards.ieee.org/getieee802/download/802.11i-2004.pdf>.
5. Hamdy S. Soliman, Mohammed Omari. *Application of synchronous dynamic encryption system in mobile wireless domains*. In Proceedings of the 1st ACM international workshop on Quality of service and security in wireless and mobile networks (Q2SWinet '05), Pages 24-30, October 2005.
6. IEEE Std. 802.1X-2004, *IEEE Standard for Local and metropolitan area networks - Port-Based Network Access Control* June, 2001. <http://standards.ieee.org/getieee802/download/802.1X-2004.pdf>
7. B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, H. Levkowitz, Ed., *Extensible Authentication Protocol (EAP)*, RFC 3748, June 2004.
8. C. Rigney, S. Willens, A. Rubens, W. Simpson, *Remote Authentication Dial In User Service (RADIUS)*, RFC 2865, June 2000.
9. P. Calhoun, J. Loughney, E. Guttman, G. Zorn, J. Arkko, *Diameter Base Protocol*, RFC 3588, September 2003.
10. D. Whiting, R. Housley, N. Ferguson, *Counter with CBC-MAC (CCM)*, RFC 3610, September 2003.
11. L. Zhou and Z. J. Haas. Securing Ad hoc networks, IEEE Network Magazine, 13(6), November/December 1999.

12. H. Luo and S. Lu. Ubiquitous and robust authentication services for ad hoc wireless networks, Proceedings of 7th IEEE Symposium on Computers and Communications, July 2002.
13. Keoh, S. L. and Lupu, E. 2002. Towards flexible credential verification in mobile ad-hoc networks. In Proceedings of the Second ACM international Workshop on Principles of Mobile Computing Toulouse, France, October 2002. POMC '02.
14. J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang, "Providing robust and ubiquitous security support for MANET," in Proc. IEEE ICNP, 2001, pp. 251-260.
15. M. Junaid, Muid Mufti, M. Umar Ilyas, *Vulnerabilities of IEEE 802.11i Wireless LAN CCMP Protocol*, Transactions on Engineering, Computing and Technology V11, February 2006.
16. Aboudagga, N., Refaei, M. T., Eltoweissy, M., DaSilva, L. A., and J. Quisquater. Authentication protocols for ad hoc networks: taxonomy and research issues. In Proceedings of the 1st ACM international Workshop on Quality of Service and Security in Wireless and Mobile Networks (Q2SWinet '05). Montreal, Quebec, Canada, October 2005.
17. Hongmei Deng; Mukherjee, A.; Agrawal, D.P., *Threshold and identity-based key management and authentication for wireless ad hoc networks*, In proceedings of International Conference on Information Technology: Coding and Computing (ITCC 2004). pages 107-111 Vol. 1, April 2004.
18. Changhua He, John C Mitchell, *Analysis of the 802.11i 4-Way Handshake*, WiSE'04, Philadelphia, Pennsylvania, USA, October 2004.
19. Arunesh Mishra, William A. Arbaugh, *An Initial Security Analysis of the IEEE 802.1X Standard*, Technical Report CS-TR-4328, Department of Computer Science, University of Maryland. February 2002. <https://drum.umd.edu/dspace/handle/1903/1179?mode=full>
20. W. A. Arbaugh. An inductive chosen plaintext attack against WEP/WEP2. IEEE Document 803.11-01/230, May 2001.
21. W. A. Arbaugh, N. Shankar and Y. J. Wan. Your 802.11 wireless network has no clothes. In Proceedings of IEEE International Conference on Wireless LANs and Home Networks, December 2001.
22. N. Borisov, I. Goldberg and D. Wagner. Intercepting mobile communications: the insecurity of 802.11. In proceedings of ACM International Conference on Mobile Computing and Networking, July 2001.
23. R. Housley, W. Polk, W. Ford and D. Solo. Internet X.509 public key infrastructure certificate and certificate revocation list (CRL) profile. Internet Request for Comments (RFC 3280), April 2002.
24. Yi-an Huang, Wei Fan, Wenke Lee, Philip S. Yu. Cross-feature analysis for detecting ad-hoc routing anomalies. Proceedings. 23rd International Conference on Distributed Computing Systems, Pages: 478 487, May 2003.
25. Claude Crepeau and Carlton R. Davis. A certificate revocation scheme for wireless ad hoc networks. Proceedings of first ACM Workshop Security of Wireless Ad hoc and Sensor Networks, 2003.

26. T. Wu, Yuan Xue and Y. Cui. Preserving Traffic Privacy in Wireless Mesh Networks. Proceedings of WoWMoM'06, 2006.
27. Souheila Bouam and Jalel Ben Othman. Data security in ad hoc networks using multipath routing, Proceedings of 14th IEEE International Symposium on Personal, Indoor and Mobile Radio Communication, 2003.
28. S. Michell and K. Srinivasan. State based key hop protocol: A lightweight security protocol for wireless networks. Proceedings of PE-WASUN'04, October 2004.
29. Srdjan Capkun, Levente Buttyan and Jean-Pierre Hubaux. Self-organized public-key management for mobile Ad Hoc networks. IEEE transactions on mobile computing, vol 2, no 1, 2003.
30. R. Housley, *Using Advanced Encryption Standard (AES) Counter Mode With IPsec Encapsulating Security Payload (ESP)*, RFC 3686, January 2004.