A Secure Hybrid Wireless Mesh Protocol for 802.11s Mesh Network*

Md. Shariful Islam, Young Jig Yoon, Md. Abdul Hamid,
and Choong Seon Hong**

Department of Computer Engineering, Kyung Hee University, Republic of Korea
{sharif,yjyoon,hamid}@networking.khu.ac.kr, cshong@khu.ac.kr

Abstract. Wireless Mesh Network (WMN) has emerged as a key technology and found a great deal of interest for the researchers in the recent past. Hybrid Wireless Mesh Protocol (HWMP) is the default path selection (i.e., routing) protocol fully specified in the current draft D.1.06 of 802.11s for WMN. However, security in routing or forwarding functionality is not specified in the standard. As a consequence, HWMP in its current from is vulnerable to various types of routing attacks. In this paper, we propose a secure version of HWMP (SHWMP) that operates similarly to that of HWMP but uses cryptographic extensions to provide authenticity and integrity of routing messages and prevents unauthorized manipulation of mutable fields in the routing information elements. We have shown through analyses and simulation that SHWMP is robust against identified attacks and provides higher packet delivery ratio and incurs little computational and storage overhead to ensure security.

Keywords: Wireless Mesh Network, Secure Hybrid Wireless Mesh Protocol (SHWMP), Authentication, Merkle Tree.

1 Introduction

Wireless Mesh Networks (WMN) is gradually maturing to a point where it cannot be ignored when considering various wireless networking technologies for deployment. It has emerged as a key technology to support a numerous number of application scenarios like broadband home networking, community and neighbourhood networking, enterprise networking, metropolitan area networking etc. with its unique characteristics like self-configuring capability, lower cost, robustness and easy network maintenance[1]. The increased interest in WMN has reflected in producing a standard named IEEE 802.11s, which is in progress and expected to be finalized by mid 2009. The current draft standard D1.06 [2] of 802.11s is the first standard that introduces the concept of embedding routing in layer-2. Interoperability between devices of different vendors is the key reason for integrating routing in MAC layer.

* This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD)" (KRF-2006-521-D00394).

** Corresponding author.

O. Gervasi et al. (Eds.): ICCSA 2008, Part I, LNCS 5072, pp. 972–985, 2008.

The network architecture of a 802.11s WMN is depicted in Fig. 1. A mesh point (MP) is an IEEE 802.11s entity that can support WLAN mesh services. A mesh access point (MAP) is an MP but can also work as an access point. A mesh portal (MPP) is a logical point that connects the mesh network to other networks such as a traditional 802.11 WLAN or a non-802.11 network. The current 802.11s standard defines secure links between MPs, but it does not provide end-to-end security. The security in routing or forwarding functionality is not also specified in 802.11s [3]. Our study identifies that HWMP is vulnerable to various types of routing attacks like flooding, route re-direction, spoofing etc. The main reason is that intermediate nodes need to modify mutable fields (i.e., hop count, TTL, metric etc) in the routing element before forwarding and re-broadcasting them. Since other nodes will act upon those added information, these must also be protected somehow from being forged or modified. However, only source authentication does not solve this problem, because the information are added or modified in intermediate nodes. This motivates us to include hop-by-hop authentication in our proposal.

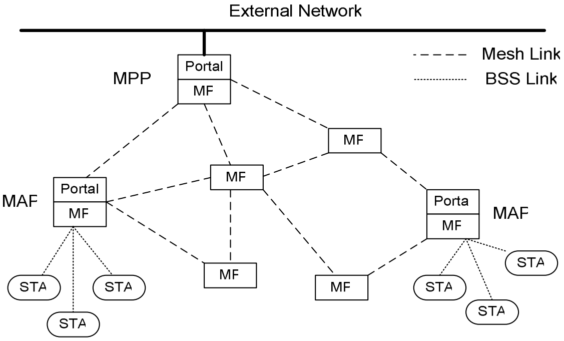


Fig. 1. Network architecture of 802.11s

We propose a secure routing protocol for 802.11s which takes into consideration the existing key hierarchy of 802.11s, identifies the mutable and non-mutable fields in the routing message, protects the non-mutable part using symmetric encryption and authenticates mutable information using Merkle tree [4].

The remainder of the paper is organized as follows. Following section discusses some of the related works. Section 3 briefly introduces HWMP in 802.11s. Possible attacks are shown in section 4. Section 5 shows the key distribution in 802.11s. We have proposed our idea in section 6 followed by security analysis in section 7. We have evaluated network performance through simulation in section 8. Finally, section 9 concludes our work.

2 Related Research

Research on layer-2 routing is still in its early age. As of now, there is no state-of-the-art solution exists in the literature for securing layer-2 routing. In [5], the authors have just summarized the proposed routing from 802.11s draft. Ref [6] describes just an update of layer-2 routing in the current draft. 802.11s's framework and research challenges are summarized in [3].

Apart from these, there has been some research on securing layer 3 routing. Ariande in [7] ensures a secure on-demand source routing. Authentication is done using TESLA [8], digital signatures and standard MAC. However, as the route request is not authenticated until it reaches the destination, an adversary can initiate route request flooding attack. A variant of Ariande named endairA is proposed in [9] that with a difference that instead of signing a route request, intermediate nodes sign the route reply. It requires less cryptographic computation, but still vulnerable to malicious route request flooding attack. SAODV [10] is a secure variant of AODV. Operations are similar to AODV, but uses cryptographic extensions to provide authenticity and integrity of routing message. It uses hash chains in order to prevent manipulation of hop count field. However, an adversary can always increase the hop count. Another secure on-demand distant vector protocol, ARAN (Authenticate Routing for Ad hoc Networks), is presented in [11]. Just like SAODV, ARAN uses public key cryptography to ensure integrity of routing message. However, a major drawback of ARAN is that it requires extensive signature generation and verification during the route request flooding.

In our proposed scheme, we will use the existing keying hierarchy specified in current 802.11s specification. So, there is no extra burden for enforcing external keying mechanism (like PKI, KDC etc.). That is, we are not assuming that a pairwise key exists between any two nodes in the networks as path security can not be assured in 802.11s. The Group Transient Key (GTK) is used for encrypting broadcast message whereas Pairwise Transient Key (PTK) is used for encrypting unicast message. We have used Merkle Tree [4] in our scheme for authenticating mutable fields in the routing information elements. Our secure routing employs symmetric cryptographic primitives only and does not assume the existence of pairwise shared key for source-destination. Rather, a Merkle tree-based hop-by-hop authentication mechanism is devised exploiting existing key-hierarchy of 802.11s standard.

3 Overview of HWMP in 802.11s

HWMP, referred as the path selection protocol in 802.11s has combines the flavor of reactive and proactive strategy by employing both on-demand path selection mode and proactive tree building mode.

On-demand mode allows two MPs to communicate using peer-to-peer paths. This mode is mainly used by nodes that experience a changing environment and when there is no root MP configured. On the other hand, proactive tree building mode can be an efficient choice for nodes in a fixed network topology. The mandatory routing metric used in HWMP is the airtime cost metric [2] that measures the link quality (e.g. amount of channel resource consumed by transmitting a frame over a particular link). In HWMP, both on demand and proactive mode can be used simultaneously.

In an On-demand mode a source MP broadcast *path request* (PREQ) message requesting a route to the destination. The PREQ is processed and forwarded by all intermediate MPs and sets up the reverse path from the destination to the source of route discovery. The destination MP or any intermediate MP with a path to the destination may unicast a *path reply* (PREP) to the source MP that creates the forward path to the destination.

Whereas in *Proactive Tree Building* mode, the MP that configured as a root MP (i.e usually the MPP) can initiate route discovery process in two ways. *Firstly,* it announces its presence by periodically broadcasting a *root announcement* RANN message that propagates metric information across the network. Upon reception of a RANN message, an MP that has to create or refresh a path to the root MP sends a unicast PREQ to the root MP. The root MP then unicast a PREP in response to each PREQ. The unicast PREQ creates the reverse path from the root MP to the originating MP, while the PREP create the forward path from the MP to the root MP. *Secondly*, the root MP proactively disseminates a PREQ message to all the MPs in the networks with intent to establish a path. An MP after receiving a proactive PREQ, creates or updates its path to the root MP by unicasting a Proactive PREP, if and only if the PREQ contains a greater sequence number, or the sequence number is the same as the current path and the PREQ offers a better metric than the current path to the root MP.

HWMP also allows both on-demand and proactive mode to work simultaneously. This hybrid mode is used in situations where a root MP is configured and a mesh point S want to send data to another mesh point D but has no path to D in its routing table. Instead if initiating on demand mode, S may send data to the root portal, which in turns delivers the data to D informing that both S and D are in the same mesh. This will trigger an on-demand route discovery between S and D and subsequent data will be forwarded using the new path. A more detailed description regarding existing HWMP can be found in [2][5][6].

4 Attack Scenarios

Following subsections briefly discuss the possible attacks that can be launched in HWMP for path selection.

4.1 PREQ Flooding

The simplest of attack that a malicious node can launch is by flooding the network with a PREQ messages destined to an address which is not present in the network. As the destination node is not present in the network, every intermediate node will keep forwarding the PREQ message. As a result, a large number of PREQ message in a short period will consume the network bandwidth and can degrade the overall throughput.

4.2 Route Re-direction

A malicious node can launch a route re-direction attack by modifying mutable fields in the control packets such as hop count, sequence number and metric field. A malicious node M can divert traffic to itself by advertising a route to a destination with a Destination Sequence Number (DSN) greater than the one it received from the destination. For example, the malicious node M in the Fig. 2a receives a PREQ from A which was originated from S for a route to node D. As HWMP allows intermediate PREP, M can unicast a PREP to A with a higher destination sequence number than

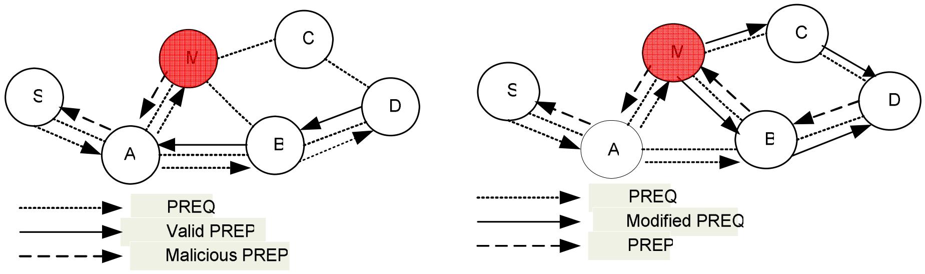


Fig. 2. Route re-direction attack. a) increasing DSN, b)decreasing metric.

the value last advertised by D. After getting the PREQ message D will also unicast PREP to the source S. At some point, A will receive both the PREPs and consider the PREP with higher destination sequence number as the valid one and discards the original PREQ as if it was stale. So, A will re-direct all subsequent traffic destined for D to the malicious node M.

Route re-direction attack can also be possible by modifying the metric field used in the HWMP PREQ messages. Value of the metric field is used to decide the path from a source to a destination, which is a cumulative field contributed by each node in the path. A malicious node can modify the mutable metric field to zero to announce a better path to a destination. As depicted in Fig. 2b, M can modify the metric field in the PREQ to zero and re-broadcasts it to the network. So, the reverse path created should go through the malicious node M. As a result, all traffics to the destination D will be passed through the attacker.

4.3 Formation of Routing Loops

A malicious node can create routing loop [11] in a mesh network by spoofing MAC addresses and modifying the value of the metric field. Consider the following network (Fig. 3) where a path exists between the source S and destination X that goes through node B and C. Also, there is a path exists from A to C through D. Assume that a malicious node M as shown in Fig. 3a, is present in the vicinity where it can listen to the PREQ/PREP messages pass through A,B,C and D during route discovery process. It can create a routing loop among the nodes A, B, C and D by impersonation combined with modification of metric field in PREP message.

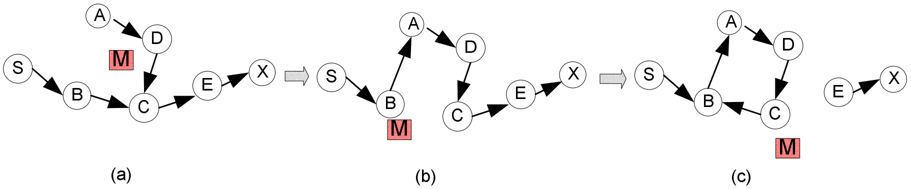


Fig. 3. Formation of Routing loops

First it impersonates node A's MAC address and moved out of the reach of node A and closer to node B. And then it sends a PREP message to node B indicating a better metric value than that of the value received from C. So, node B now re-establishes its route to X that should go through A as shown in Fig. 3b. At this point, the malicious node impersonates node B and moves closer to node C and sends a PREP to node C indicating a better metric than the one received from E. So, node C will now choose B as the next hop for its route to the destination X as shown in Fig. 3c. Thus a loop has been formed and the destination X is unreachable from all the four nodes.

5 Key Distribution in 802.11s

802.11s ensures link security by using Mesh Security Association (MSA) services. 802.11s extends the security concept of 802.11 by a key hierarchy, inherits functions of 802.11i and uses 802.1X for initial authentication. The operation of MSA relies on mesh key holders, which are functions that are implemented at MPs within the WMN.

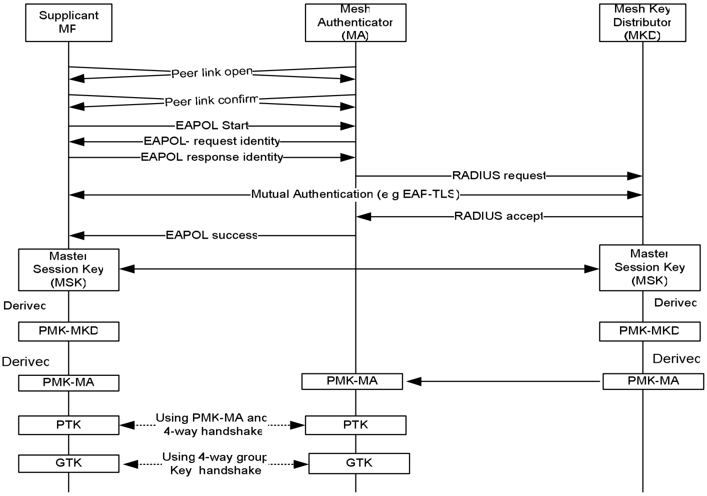


Fig. 4. Key establishment and authentication mechanism in 802.11s

Two types of mesh key holders are defined: mesh authenticators (MAs) and mesh key distributors (MKDs). A single MP may implement both MKD and MA key holders, an MA alone and no key holders [2]. Fig. 4 depicts the key establishment and authentication process of 802.11s.

The first level of link security branch, PMK-MKD is mutually derived by the supplicant MP and MKD, from the Master Session Key (MSK) that is created after the initial authentication phase between supplicant MP and MKD or from a pre-shared key (PSK) between MKD and supplicant MP, if exists. The second level of link security branch PMK-MA is also derived by the supplicant MP and MKD. MKD then delivers the PMK-MA to the MA and thus permits to initiate MSA 4-way handshake which results in deriving a PTK of 512 bits between supplicant MP and MA.

During the MSA 4-way handshake MA receives the GTK of the supplicant MP. After the completion of MSA-4way handshake, a group handshake is used to send the GTK of the MA to the supplicant MP. The GTK is a shared key among all supplicant MPs that are connected to the same mesh authenticator (MA). In our proposed secure routing algorithm, PTK is used for encryption of unicast messages and GTK is used for encrypting broadcast messages.

6 Proposed Secure HWMP

The routing protocol proposed in this section is a secure version of Hybrid Wireless Mesh Protocol (SHWMP). As specified in [2], HWMP routing information elements have a mutable and a non-mutable part. We exploit these existing mutable and non-mutable fields to design a secure layer-2 routing. More specifically, we (i) use the existing key distribution, (ii) identify the mutable and non-mutable fields, (iii) show that mutable fields can be authenticated in hop-by-hop fashion using the concept of Merkle tree, and (iv) use symmetric encryption to protect non-mutable fields. The rest of this section describes the proposed protocol in details.

6.1 Use of Keys

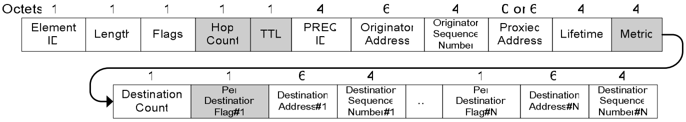
All the entities in the mesh infrastructure (MP, MAP and MPP) can act as supplicant MP and Mesh Authenticator (MA). Before initiating a route discovery process, all the MPs authenticate its neighboring MPs, send its GTK and establish PTK through key distribution process described in Section 5. We use this GTK for securing broadcast messages such as PREQ, RANN and PTK is used to secure unicast messages such as PREP, proactive PREQ.

6.2 Identification of Mutable / Non-mutable Fields

The information elements in the HWMP contain fields that can be modified in the intermediate routers which we termed as mutable and those that can not be modified termed as non-mutable fields.

Fig. 5 shows the format of a PREQ element where the mutable fields are:

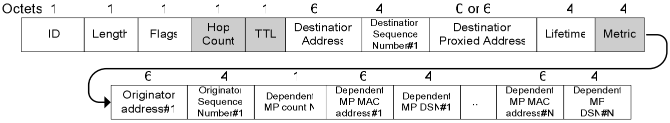
a. Hop count field: Provides information on the number of links in the path, incremented by each intermediate node, but it is not used for routing decision.
b. TTL field: The time to leave field defines the scope of the PREQ in number of hops. TTL value is decremented by 1 at each intermediate node.
c. Metric field: HWMP uses an airtime link metric instead of hop count metric as in AODV, to take a decision on path selection. Whenever an intermediate node receives a PREQ that is to be forwarded, it calculates the airtime cost to the current path and adds the value to the existing metric field.
d. Per destination flag: The Destination Only (DO) and Reply and Forward Flag (RF) determine whether the route-reply message (RREP) will be sent by intermediate node or only by destination. If DO flag is not set and RF flag is set, the first intermediate node that has a path to the destination sends PREP and forwards the PREQ by setting the DO flag to avoid all intermediate MPs sending a PREP. In this case, per destination flag field is also a mutable field.



Octets	1	1	1	1	1	4	6	4	0 or 6	4	4
	Element ID	Length	Flags	Hop Count	TTL	PREQ ID	Originator Address	Originator Sequence Number	Proxied Address	Lifetime	Metric

1	1	6	4		1	6	4
Destination Count	Per Destination Flag#1	Destination Address#1	Destination Sequence Number#1	..	Per Destination Flag#N	Destination Address#N	Destination Sequence Number#N

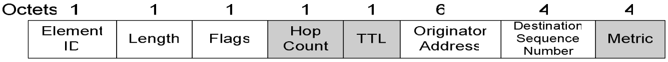
Fig. 5. Format of a PREQ element



Octets	1	1	1	1	1	6	4	0 or 6	4	4
	ID	Length	Flags	Hop Count	TTL	Destination Address	Destination Sequence Number#1	Destination Proxied Address	Lifetime	Metric

6	4	1	6	4		6	4
Originator address#1	Originator Sequence Number#1	Dependent MP count N	Dependent MP MAC address#1	Dependent MP DSN#1	..	Dependent MP MAC address#N	Dependent MP DSN#N

Fig. 6. Format of a PREP element



Octets	1	1	1	1	1	6	4	4
	Element ID	Length	Flags	Hop Count	TTL	Originator Address	Destination Sequence Number	Metric

Fig. 7. Format of a RANN element

Fig. 6 and Fig. 7 show the format of a PREP and RANN information element. In both the cases, the mutable fields are hop-count, TTL and metric indicated by shadowed boxes.

6.3 Construction of Merkle Tree

Let the mutable fields of routing information elements that need to be authenticated are v_1, v_2, v_3 and v_4. We hash each value v_i into u_i with a one-way hash function such that $u_i = h(v_i)$. Then we assign the hash values to the leaves of the binary tree as shown in Fig. 8.

Moreover, to each internal vertex u of this tree, we assign a value that is computed as the hash of the values assigned to the two children of u such as $u_{12} = h(u_1 \| u_2)$. Finally, we found the value of the root and make a message authentication code on

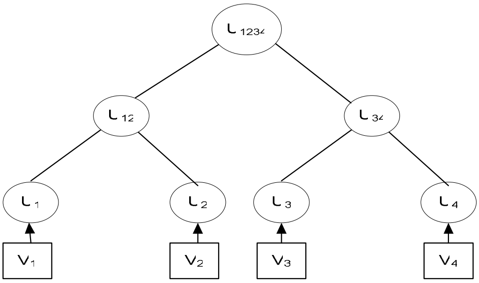


Fig. 8. Construction of Merkle tree

the root by using GTK of the sender or by PTK between sender and receiver for authenticating broadcast and unicast messages, respectively. The sender can reveal a value v_i that needs to be authenticated along with the values assigned to the siblings of the vertices along the path from v_i to root that we denote as authentication path, *authpath(* v_i *)*. The receiver can hash the values of the authentication path in appropriate order to compute the root and compare the value assigned to the root. If these two values match, then the receiver can be assured that the value v_i is authentic.

6.4 Securing on Demand Mode

Consider a source MP S in Fig. 9 which wants to communicate with a destination MP X.

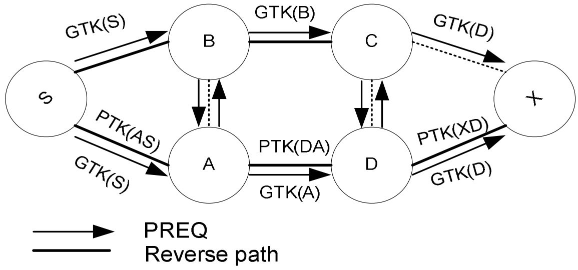


Fig. 9. Secure on-demand path selection

In order to establish a secure route, source node S, destination node X and set of intermediate nodes F_1 that include A, B and F_2 that includes C, D executes the route discovery process in the following way:

$$S \rightarrow * : MAC_{GTK} root(S), \{v_i, authpath(v_i)\}, \{PREQ - MF\}_{GTK} \quad (1)$$

$$F_1 \rightarrow * : MAC_{GTK} root(F_1), \{v_i, authpath(v_i)\}, \{PREQ - MF\}_{GTK} \quad (2)$$

$$F_2 \rightarrow * : MAC_{GTK} root(F_2), \{v_i, authpath(v_i)\}, \{PREQ - MF\}_{GTK} \quad (3)$$

$$X \rightarrow F_2 : MAC_{PTK}^{X,F_2} root(X), \{v_i, authpath(v_i)\}, \{PREP - MF\}_{PTK}^{X,F_2} \quad (4)$$

$$F_2 \rightarrow F_1 : MAC_{PTK}^{F_2,F_1} root(F_2), \{v_i, authpath(v_i)\}, \{PREP - MF\}_{PTK}^{F_2,F_1} \quad (5)$$

$$F_1 \rightarrow S : MAC_{PTK}^{F_1,S} root(F_1), \{v_i, authpath(v_i)\}, \{PREP - MF\}_{PTK}^{F_1,S} \quad (6)$$

From the key management of 802.11s, the node S is equipped with one GTK that it shares with its neighbors and set of PTKs for communicating with each neighbor individually. Before broadcasting the PREQ, it first creates a Merkle tree with the leaves being the hash of mutable fields of PREQ message. S then creates a MAC on the root of the Merkle tree it just created. Then, S broadcasts message (1) which includes the MAC of the root created using the GTK, mutable fields v_is that need to be authenticated along with the values of its authentication path *authpath(* v_i *)*and encrypted PREQ message excluding the mutable fields.

Any of the neighboring nodes of S, after receiving the PREQ, tries to authenticate the mutable fields by hashing the values received in an ordered way, create a MAC on it using the shared GTK and comparing that with the received MAC value of the root. If the two values match, the intermediate MP is ascertain that the values are authentic and come from the same source that created the tree.

Let us consider for example that B receives a PREQ from its neighboring node S and wants to authenticate the value of the metric field M as shown in Fig 10. According to our protocol, B and C should receive the value M along with the values of the authentication path of M in the Merkle tree such as U_H and U_{TF}. B and C can now verify the authenticity of M by computing $h(h(h(M) \| U_H) \| U_{TF})$ and a MAC on this value using the key GTK. It then compares the received MAC value with the new one, if it found a match, then it can assure that the value M is authentic and came from the same entity that has created the tree and computed the MAC on the U_{root}.

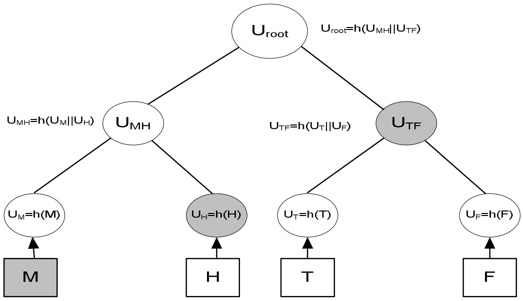


Fig. 10. Authentication path for the Metric field M in Merkle Tree

The intermediate nodes then update the values of the mutable fields like hop count, metric and TTL and create Merkle trees from the modified fields. They also decrypt the non-mutable part of the PREQ message and re-encrypt it with their own broadcast key and re-broadcast (as shown in (2) and (3)) the PREQ message following the same principle. After receiving the PREQ, the destination MP updates the mutable fields; creates its own Merkle Tree and unicasts a PREP message as (4) using same principle but this time using PTK instead of using GTK. The PREP is propagated as (5) and (6) to the source MP in the reverse path created using PREQ and thus a secure forward path from the source to the destination is established.

6.5 Securing Proactive Mode

In the *Proactive RANN* mode, the RANN message is broadcasted using the group transient key as shown in Equation (7) to (9) to protect the non-mutable fields and authenticate the mutable fields (hop count, TTL and metric) using the Merkle tree approach. As there are only three mutable fields in the RANN message a node requires generating a random number to construct the Merkle tree. After receiving the

RANN message an MP that needs to setup a path to the root MP unicast a PREQ to the root MP as per Equation (10) to (12). On receiving each PREQ the root MP replies with a unicast PREP to that node as described in Equation (13) to (15). *Proactive PREQ* mode can also be secured by transmitting proactive PREQ and PREP in the same way discussed above.

$$\mathrm{R} \rightarrow *:\ \mathrm{MAC}_{\mathrm{GTK}}\mathrm{Root(R)},\ \{\mathrm{v_i},\ \mathrm{authpath\ (v_i)}\},\ \{\text{RANN-MF}\}_{\mathrm{GTK}} \tag{7}$$

$$\mathrm{F_1} \rightarrow *:\ \mathrm{MAC}_{\mathrm{GTK}}\mathrm{Root(F_1)},\ \{\mathrm{v_i},\ \mathrm{authpath\ (v_i)}\},\ \{\text{RANN-MF}\}_{\mathrm{GTK}} \tag{8}$$

$$\mathrm{F_2} \rightarrow *:\ \mathrm{MAC}_{\mathrm{GTK}}\mathrm{Root(F_2)},\ \{\mathrm{v_i},\ \mathrm{authpath\ (v_i)}\},\ \{\text{RANN-MF}\}_{\mathrm{GTK}} \tag{9}$$

$$\mathrm{D} \rightarrow \mathrm{F_2}:\ \mathrm{MAC}_{\mathrm{PTK}}^{\mathrm{D,F_2}}\mathrm{Root(D)},\ \{\mathrm{v_i},\ \mathrm{authpath\ (v_i)}\},\ \{\text{PREQ-MF}\}_{\mathrm{PTK}}^{\mathrm{D,F_2}} \tag{10}$$

$$\mathrm{F_2} \rightarrow \mathrm{F_1}:\ \mathrm{MAC}_{\mathrm{PTK}}^{\mathrm{F_2,F_1}}\mathrm{Root(F_2)},\ \{\mathrm{v_i},\ \mathrm{authpath\ (v_i)}\},\ \{\text{PREQ-MF}\}_{\mathrm{PTK}}^{\mathrm{F_2,F_1}} \tag{11}$$

$$\mathrm{F_1} \rightarrow \mathrm{R}:\ \mathrm{MAC}_{\mathrm{PTK}}^{\mathrm{F_1,R}}\mathrm{Root(F_1)},\ \{\mathrm{v_i},\ \mathrm{authpath\ (v_i)}\},\ \{\text{PREQ-MF}\}_{\mathrm{PTK}}^{\mathrm{F_1,R}} \tag{12}$$

$$\mathrm{R} \rightarrow \mathrm{F_1}:\ \mathrm{MAC}_{\mathrm{PTK}}^{\mathrm{R,F_1}}\mathrm{Root(R)},\ \{\mathrm{v_i},\ \mathrm{authpath\ (v_i)}\},\ \{\text{PREP-MF}\}_{\mathrm{PTK}}^{\mathrm{R,F_1}} \tag{13}$$

$$\mathrm{F_1} \rightarrow \mathrm{F_2}:\ \mathrm{MAC}_{\mathrm{PTK}}^{\mathrm{F_1,F_2}}\mathrm{Root(F_1)},\ \{\mathrm{v_i},\ \mathrm{authpath\ (v_i)}\},\ \{\text{PREP-MF}\}_{\mathrm{PTK}}^{\mathrm{F_1,F_2}} \tag{14}$$

$$\mathrm{F_2} \rightarrow \mathrm{D}:\ \mathrm{MAC}_{\mathrm{PTK}}^{\mathrm{F_2,D}}\mathrm{Root(F_2)},\ \{\mathrm{v_i},\ \mathrm{authpath\ (v_i)}\},\ \{\text{PREP-MF}\}_{\mathrm{PTK}}^{\mathrm{F_2,D}} \tag{15}$$

Notations used in Equation (7) to (15) are as follows: R is considered as the root MP and D is the MP that needs to setup a path to R. F_1 and F_2 are the intermediate nodes in the path. $\mathrm{MAC_k Root(X)}$, represents the MAC of the Merkle tree's root created by node X using a shared key k. {RANN/PREQ/PREP-MF} represents the routing information elements without the mutable fields. v_i and *authpath*(v_i) denote the fields need to be authenticated and the values assigned to the authentication path from v_i to root of the tree, respectively.

7 Security and Overhead Analyses

In this section, we will analyze the proposed SHWMP in terms of robustness against the attacks presented in Section 4 and also the overhead required for ensuring secure routing.

7.1 Security Analysis

1) Preventing Flooding Attack: In the proposed SHWMP, a node can participate in the route discovery process only if it has successfully establishes a GTK and PTK through key distribution and authentication mechanism of 802.11s. Thus it will not be possible for a malicious node to initiate a route discovery process with a destination address that is not in the network. Again, as the PREQ message is encrypted during transmission, a malicious node can not insert new destination address.

2) Modification of Routing Messages: The root cause of route re-direction attacks are modification of mutable fields in routing messages. These mutable fields are authenticated in each hop. If any malicious node modifies the value of a field in transit, it will be readily detected by the next hop while comparing the new MAC with the received one. It will find a miss-match in comparing the MACs and modified packet will be discarded.

3) Avoiding Formation of Routing Loops: Formation of routing loops requires gaining information regarding network topology, spoofing and alteration of routing message. As all the routing information is encrypted between nodes, an adversary will be unable to learn network topology by overhearing routing messages. Spoofing will not benefit the adversary as it will require authentication and key establishment to transmit a message with spoofed MAC. Moreover, fabrication of routing messages is detected by integrity check. So, proposed mechanism ensures that routing loops can not be formed.

7.2 Overhead Analysis

1) Computation cost: The computation cost of a sender and receiver are defined by following equations:

$$\langle k \times h \rangle + m + e \text{ (sender)} \tag{16}$$

$$\langle a + 1 \rangle h + m + d \text{ (receiver)} \tag{17}$$

Where, *k* is the number of hash operations required to form a Merkle tree. Cost of computing a hash function is defined by *h*. *m* is the cost involved in computing the MAC of the root, whereas e and d are encryption and decryption cost. To authenticate a particular value, a receiver need to compute the root by calculating (*a*+1) hash operations, where *a* defines the number of nodes in the authentication path.

2) Communication Overhead: It is defined by the number of routing messages required to establish a secure path and defined by (18), (19) and (20).

$$(n\text{-}1) \times broadcast + h \times unicast \quad (on\text{-}demand) \tag{18}$$

$$n \times broadcast + h \times unicast \quad (practive\ PREQ) \tag{19}$$

$$n \times broadcast + 2h \times unicast \quad (practive\ RANN) \tag{20}$$

Where, *n* is the number of nodes in the network, *h* is the number of hops in the shortest path. The number of messages required for establishing a path in HWMP is same as our proposed one. So, our protocol does not incur any extra communication overhead.

3) Storage Requirements: A node needs to store the number of fields that need to be authenticated, hashed values of the Merkle tree and the MAC of the root value. So, storage requirement of a node is given by (21).

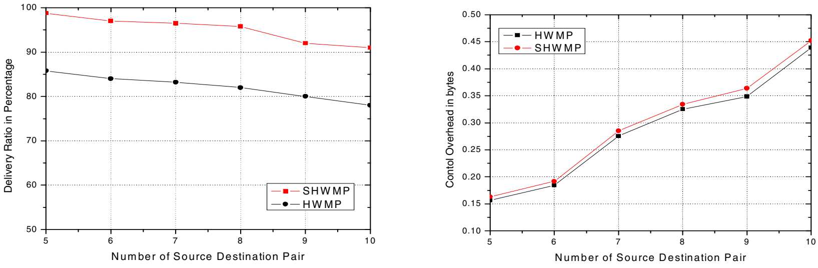
$$\sum_{i=1}^{n} d_i + (k \times l) + S_M \tag{21}$$

Where, d_i is the size of a mutable field, *k* is the number of hashes in the Merkle tree, *l* is the size of a hashed value and S_M is the size of the MAC.

8 Network Performance Analysis

We use *ns*-2 [12] to simulate our proposed secure routing (SHWMP) approach and compare that with existing HWMP. We have simulated 50 nodes in a 1500 x1500 m^2 area. We use 5 to 10 distinct source destinations pairs that are selected randomly. Traffic source are CBR (constant bit-rate). Each source sends data packets of 512 bytes at the rate of four packets per second during the simulation period of 900 seconds. We consider the following performance metrics:

1. **Packet delivery ratio:** Ratio of the number of data packets received at the destinations to the number of data packets generated by the CBR source.
2. **Control overhead (in bytes):** Ratio of the control overhead to the delivered data.



a) Packet delivery ratio

b) Control packet overhead

Fig. 11.

As shown in Fig. 11a, the packet delivery ratio is better in SHWMP than that of HWMP. Since the misbehaving nodes participates in the route discovery process, in HWMP sometimes packets are intentionally dropped by the malicious nodes. But, in the proposed protocol, malicious node can not participate in the route discovery process and thus always achieve a higher packet delivery ratio.

Fig. 11b shows that the control packet overhead for the two protocols are almost identical though SHWMP has a little more overhead as it needs to include MAC values along with the routing messages, whereas the number of control packets transmitted by the two protocols is roughly equivalent.

9 Conclusion and Future Works

The goal of this paper is to develop a secure routing mechanism for wireless mesh networks. We have proposed SHWMP, a secure extension of L2 routing specified in 802.11s. Our proposed mechanism takes into consideration the existing key hierarchy of 802.11s (so, there is no extra keying burden), identifies the mutable and non-mutable fields in the routing message, protects the non-mutable part using symmetric encryption and uses Merkle-tree approach to authenticate mutable information. We have shown that our protocol is robust against identified attacks and computationally efficient as it uses only symmetric key operations. One of our key research goal is to develop a secure interference aware L2 routing for 802.11s wireless mesh networks.

References

1. Akyildiz, I.F., Wang, X., Wang, W.: Wireless mesh networks: a survey. Computer Networks 47(4) (March 2005)
2. IEEE 802.11s Task Group, Draft Amendment to Standard for Information Technology Telecommunications and Information Exchange Between Systems – LAN/MAN Specific Requirements – Part 11: Wireless Medium Access Control (MAC) and physical layer (PHY) specifications: Amendment: ESS Mesh Networking, IEEE P802.11s/D1.06 (July 2007)
3. Wang, X., Lim, A.O.: IEEE 802.11s wireless mesh networks: Framework and Challenges. AdHoc Networks, 1–15 (2007) doi:10.1016/j.adhoc.2007.09.003
4. Merkle, R.C.: A certified digital signature (subtitle: That antique paper from 1979). In: Brassard, G. (ed.) CRYPTO 1989. LNCS, vol. 435, pp. 218–238. Springer, Heidelberg (1990)
5. Bahr, M.: Proposed Routing for IEEE 802.11s WLAN Mesh Networks. In: 2nd Annual International Wireless Internet Conference (WICON), Boston, MA, USA (August 2006)
6. Bahr, M.: Update on the Hybrid Wireless Mesh protocol of 802.11s. In: Proc. of IEEE International Conference on Mobile Adhoc and Sensor Systems, MASS 2007, pp. 1–6 (2007)
7. Hu, Y.-C., Perrig, A., Johnson, D.B.: Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc Networks. In: Proc. MobiCom 2002, Atlanta, GA, September 23–28 (2002)
8. Perrig, A., Canetti, R., Tygar, J.D., Song, D.: Efficient authentication and signing of multicast streams over lossy channels. In: Proc. of IEEE Symposium on Security and Privacy, pp. 56–73 (2000)
9. Gergely, L.B., Vajda, I.: Provably secure on-demand routing in Mobile Adhoc Networks. IEEE transactions on Mobile Computingm 5(11), 1533–1546 (2006)
10. Zapata, M.G., Asokan, N.: Securing Adhoc rouring protocols. In: Proc. of ACM Workshop of Wireless Security (Wise), September 2002, pp. 1–10 (2002)
11. Sangiri, K., Dahil, B.: A Secure Routing Protocol for Ad Hoc Networks. In: Proc. of 10th IEEE International Conference on Network Protocols (ICNP 2002) (2002)
12. The Network Simulator – ns-2, http://www.isi.edu/nsnam/ns/index.html