# Multi-path Routing Scheme for Preserving Privacy in Wireless Mesh Networks

Cao Trong Hieu, Choong Seon Hong
Department of Computer Engineering, Kyung Hee University
hieuct@networking.khu.ac.kr, cshong@khu.ac.kr

## Abstract

Enhancing security for routing in Multi-hop Wireless Mesh Networks currently becomes challenging topic because of inherent vulnerabilities of wireless communications. To utilize the characteristics of WMN's topology, in this paper, we propose an algorithm to preserve privacy for routing. This idea comes from the fact that if we can separate data traffic into more than one path, the probability to capture all traffic from intermediate node is very small. It means it is very difficult to launch traffic analysis attacks because of traffic confidentiality. In addition, a new technique to securely hide the real source and destination addresses is proposed along with an Adaptive Key Agreement Scheme. We apply Information Entropy to model our routing traffic and highlight the robustness of the algorithm. We also present a detail traffic evaluation observed from neighboring nodes to show the availability of our proposal in term of loop free and computational overhead.

***Index Terms***
Security, Routing, Privacy Preservation, Information Entropy, Wireless Mesh Network.

## 1. Introduction

Along with Mobile Ad-hoc Network, Wireless Mesh Network recently has attracted increasing attention thank for the low-cost deployment and topology flexibility [5]. WMN represent a good solution to providing wireless Internet connectivity in a large scale. This new and promising paradigm allows for deploying network at much lower cost than with classic WiFi network. However, multi-hop makes routing in WMNs a very important and necessary functionality of the network. Thus, the routing mechanism must be secured.

We consider a Mesh Topology shown in Fig. 1. In this network, multiple mesh routers communicate with each other to form a multi-hop wireless backbone that forwards user traffic to the gateways which are connected to the Internet. Client devices access a stationary wireless mesh router at its residence

Confidentiality (privacy) is one of the most important criteria regarding security aspect. Despite the necessity, limited research has been conducted towards privacy preservation in WMN. In this paper, we focus on traffic confidentiality which prevents the traffic analysis attack from the mesh router. Our target is designing a lightweight traffic privacy preserving mechanism for WMN which is able to balance between traffic analysis resistance and bandwidth cost.

The key idea is if the traffic between source S and destination D goes through only one route, any intermediate node can easily observe the entire traffic between S and D. This route is vulnerable to traffic privacy attacks. To tackle this weakness, we propose a Multi-path routing mechanism which utilizes multiple paths for data delivery and can protect attacks based on data analysis. When the data is transmitted by more than one route, it is very difficult for attackers to discover the entire route of hope-to-hop communication. It means they can not completely collect data from source and destination, thus they can not restore and understand the meaning of stolen data.

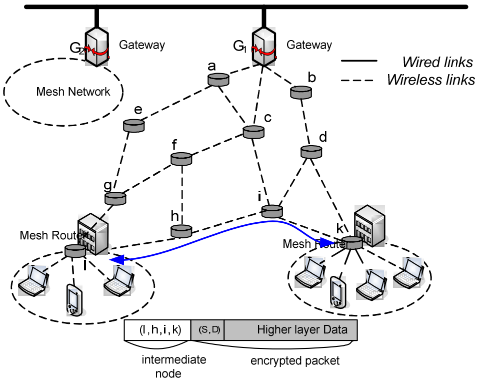


**Figure 1: General Mesh Topology**

The rest of the paper is organized as follows: Section 2 briefly discusses some related works. In Section 3, we propose an algorithm to find the multi-path between two mesh routers (nodes) when end-users want to communicate with each other or access to Internet. In this section, we

* ***This work was supported by MIC and ITRC project.***
***Dr. CS Hong is corresponding author.***

focus on traffic confidentiality and solve problem of traffic pattern concealment. In section 4, we propose an Adaptive Key Agreement Scheme to encrypt the data packets and transmit through multiple disjoint paths found in the previous step. In our scheme, we introduce a new technique that can hide real source and destination addresses. To make our proposal more reliable, we apply Information Entropy to model our routing traffic and prove the robustness of the algorithm in Section 5. Finally, section 6 exposes some perspectives for further work.

## 2. Related Work

WMN is a hybrid network which has both mobile parts and stationary parts. However, due to limited capacity, delay constrains [2, 3] and the lack of security guarantees [4], WMNs are not yet ready for wide-scale deployment. The first problem can be solved by using multi-radio and multi-channel Transit Access Points (TAPs) [5]. The other most important challenge concerned here is security especially in routing protocol.

In the existing literature, traffic padding [7] and anonymous overlay routing [6] have been proposed to preserve user traffic privacy and increase the difficulty for traffic analysis. The onion routing [9] developed by David Goldschlag et al. can secure communication through an unpredictable path but it is necessary to encrypt message between routers. This means all intermediate nodes have to involve in encryption/decryption process which cause more overhead. In wireless ad-hoc networks, authors proposed schemes for location and identity privacy in [8]. However, none of them can be applied to WMN directly and they also require a large amount of time, memory capacity to process. Our proposed routing protocol will solve those existing constrains.

In reality, the traffic of a node is a continuous function of time, as shown in Fig. 2. However, in our proposal, to apply Information Entropy for privacy preservation, we consider the traffic as discrete random variable. Therefore, as the first step, we discrete the continuous traffic into piece-wise approximation of discrete values. Then we measure the amount of traffic in each period, usually in terms of number of packets, with assumption that the packet sizes are all equal.

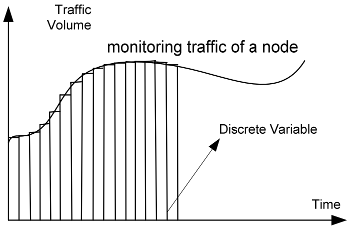


**Figure 2: Sampling continuous traffic**

In the routing table, as shown in Fig. 1, the Source and Destination's addresses are encrypted by existing encryption schemes. The intermediate nodes only know the address of Mesh Routers in source and destination residence. This provides the second security layer to preserve privacy.

## 3. Multi-path Finding Algorithm

To apply our algorithm to routing protocol, some pre-conditions are established and require a little bit change in routing table. We define *Found Route* to count and keep the number of paths found after the algorithm is executed. *Node Occupied Status* is 0 at initial stage and is set to 1 if a node is not available or it is already in a path. *Number_RREQ* is the number of requests sent from source to destination. Each time a route is found or *Request_Time* is over, the source will send another request and *Number_RREQ* will be counted down. In our algorithm, *Number_RREQ* is equal to the number of neighbors of source node. *Request_Time* is adjustable value. Its value can be flexibly assigned. It is not too long to avoid overhead and not too short to guarantee path finding process.

```
Initial
Node's Occupied Status = 0;Found Route = 0;
Number_RREQ = n  /* n is the number of neighbors of source node*/
Request_time = k;
Step 1: flood $RREQ_S$ to unoccupied neighbor nodes;
check node's availability & $arrived\_HC_i$;
Step 2: if $arrived\_HC_i < Current\_HC_i$
    { if Node_ Add == Destination_Add
      {Found Route ++; Set Occupied_status = 1;
        Number_RREQ --};
      else return Step 1};
        else { discast $RREQ_i$;
                Set Occupied_status = 1; finish};
Step 3: repeat step 1;
        finish while { Number_RREQ = =0 or
Request_time == 0}
```

*Hop count (HC)* is used to determine the shortest path and it is increased by 1 if *RREQ* or *RREP* is forwarded each hop. In this algorithm, *HC* is also used to avoid *RREQ's* loop back which also causes time and energy consumption. In *Step 1*, all node states are unoccupied. The *RREQ* is sent to all neighbors of source node. *Node's availability* [1] will be checked in this step. There are many criteria to decide whether a node has ability and capacity to become an intermediate node in a route. If a node has enough *Signal strength, Bandwidth, and Energy Remaining*, it can be intermediate node, otherwise, it will discard the *RREQ*, set *Node Occupied Status* = 1, notify *Source Node* of its conditions, and will not involve in the procedure. By this way, the number of nodes involving in the algorithm is limited.

As mentioned above, *Hop Count (HC)* is stored in routing table of each node and compared with new *HC* index when a *RREQ* arrives. If new *RREQ* has *HC* smaller than current one, the node will update new *HC* and go to *Step 2*.

In *Step 2*, *Node's Address* is compared with *Destination Address* in *RREQ*. If it has the same address, *Found Route* is increased, *Node Occupied Status* is set to 1 and the number of *RREQ* is decreased by 1. At this time, *Number_RREQ* and *Request_Time* are checked in *Step 3* and if one of them equals 0, the algorithm is finished. Those conditions guarantee overhead avoidance.

Note that when a node does not satisfy the condition in *Step 2*, it will uni-cast back to notify the source and from this time it will not participate in the routing process. Moreover, the repetition of step 1 in step 2 is different from step 3 because the *Number_RREQ* is not counted down. *Number_RREQ* is only counted down when a new route is found. That is the reason why we need *Request_Time* to avoid overhead.

After the finding algorithm finished, in the routing table of involved nodes, the information about the number of routes and list of nodes in each route are stored. From that information, source node starts to send data through separate paths. As we discussed in [1], the path between source and destination in this case also need not be shortest path regarding hop count.

## 4. Adaptive Key Management Scheme

As briefly mentioned in section 1, in this part, we introduce a new technique that can hide real source and destination addresses. Only nodes which have common keys can extract addresses of source and destination. Intermediate nodes will not involve in encryption/decryption process, so that the proposed scheme can avoid computational overhead.

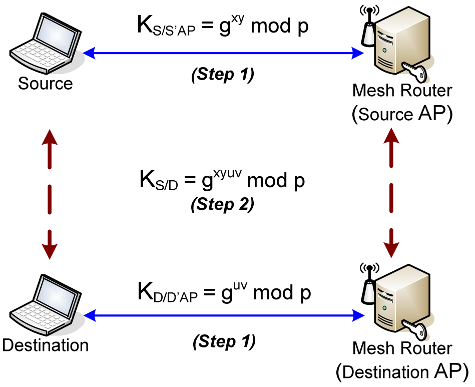


**Figure 4: Key Exchange Scenario**

After process *Multi-Path Finding Algorithm*, the source and current source AP run 2-party Diffie-Hellman in parallel with the destination and current destination AP do[10, 11].

The key exchange includes 2 steps. At the first step (represented by green arrows in Fig. 4), the source node and its access point (AP) choose a secret number *(x, y)* respectively, a large co-prime *(g, p)*, and exchange to make a common key $K_{S/S'AP}$:

$$K_{S/S'AP} = g^{xy} \bmod p$$

At this time, the destination node and its AP also choose a secret number *(u, v)* respectively and exchange to make a common key $K_{D/D'AP}$:

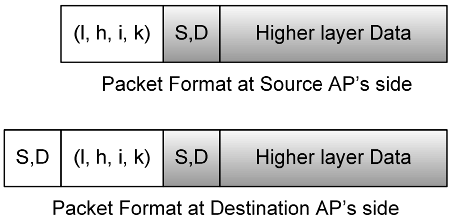
$$K_{D/D'AP} = g^{uv} \bmod p$$

In the second step (represented by red arrows in *Fig. 4*), the source AP and the destination AP run 2-party D-H in parallel with source and destination do and compute a common shared key $K_{S/D}$:

$$K_{S/D} = g^{xyuv} \bmod p$$

After this process, 4 nodes will have the same key $K_{S/D}$ and they can communicate securely.

To hide the real *S/D* addresses, we proposed a new technique that intermediate nodes can not extract to know address of *S/D*. This technique can prevent almost kinds of attacks based on data privacy.



Packet Format at Source AP's side

Packet Format at Destination AP's side

**Figure 5: Additional Field in Packet's Format**

At the source side, before transmitting, the data is split and encrypted with *S/D* addresses. After that, the addresses of intermediate nodes found in previous step (*section 3*) are attached without encryption. By this way, the intermediate nodes can only extract the source AP address and destination AP address. In the figure 5, l and k is the address of source AP and destination AP respectively. For each found route, there is a different sequence of intermediate nodes between l and k because the paths are disjoint. Each time an intermediate node receives a packet, it simply forwards this packet to the next hop in the address sequence. Without the need of knowing *S/D* address, all the packets will arrive to destination AP.

One challenge for proposed scheme is how to avoid computation overhead at receiver side because normally the destination AP will broadcast packets to all wireless clients in its range in MAC protocol. As a consequence of our proposed technique, it will require all destination AP's neighbor nodes decrypt the packets to know weather those packets are sent to them or not. To solve this problem, the destination AP will use the common key $K_{S/D}$ to extract the *S/D* address in each packets and puts it in the unencrypted part before sending to its neighbor as showed in the *figure 5*. When all clients receive the packets, they simply compare the destination address. If a packet is for a node, it can decrypt the packet thanks to $K_{S/D}$. If the packet is not for this node, it will drop and also can not try to decrypt the packet. In briefly, this technique can make the second protection layer for privacy of data not only at intermediate compromised nodes but also at receiver side. It also puts a little more computation overhead only at *S/D* access points.

To illustrate the privacy preserving and evaluate the rare probability that an attacker can capture and reassemble the data from source to destination in our algorithm, in the next section, we apply Information Entropy (also called Shannon Entropy) into our proposal.

## 5. Traffic Evaluations

In the information theory, the concept of Information Entropy (Shannon Entropy) describes how much information there is in a signal or event. In fact, this concept is applied in many fields of the information theory as well as the statistical theory. In our proposal, it is used for evaluating the traffic volume that goes through separate routing paths described above.

We discrete continuous traffic into equal-size sampling period as discussed in the *section 2*, and use *A* as the random variable of this discrete value. The probability that the random variable *A* is equal to *i* (a node receives *i* packets in a sampling period) is $P(A = i)$. Likewise, $P(B^A = j)$ is the probability that $B^A$ is equal to *j*. $(i, j \in N)$.

From those definitions, the Information Entropy of the discrete random variable *A* is

$$H(A) = \sum_{i=1}^{n} P(A=i)\log_2\left(\frac{1}{P(A=i)}\right) \quad \text{(I)}$$

$$= -\sum_{i=1}^{n} P(A=i)\log_2 P(A=i)$$

*H(A)* is a measurement of the uncertainty about the outcome of *A*. It means if the value of *A* is distributed and no value predominates, *H(A)* takes its maximum value. On the other hand, if the traffic pattern is *Constant Bit Rate (CBR)*, then $H(A) = 0$, since the number of packets at any sampling period is fixed.

Similarly, we have the entropy for $B^A$ as follows.

$$H(B^A) = -\sum_{i=1}^{n} P(B^A = i)\log_2 P(B^A = i) \quad \text{(II)}$$

$B^A$ is a random variable representing the number of packets destined to node *a* observed at node *b* in a sampling period. The purpose of this equation is to evaluate the amount of information which can be observed from a neighbor of a node. For this we can assure that in case a node and some other neighbors are compromised, they also can not capture the whole sent data.

Then we define the conditional entropy of random variable $B^A$ with respect to *A* as

$$H(A/B^A) = -\sum_{j=1}^{m} P(B^A = j)\sum_{i=1}^{n} p_{ij}\log_2 p_{ij} \quad \text{(III)}$$

in which, $p_{ij} = P(A = i|B^A = j)$ is the probability that $A = i$ given that $B^A = j$. $H(A|B^A)$ can be thought of as the uncertainty remained about *A* after $B^A$ is known. The joint entropy of *A* and $B^A$ can be shown as

$$H(A, B^A) = H(B^A) + H(A/B^A) \quad \text{(IV)}$$

The mutual information of *A* and $B^A$ which represents the information we can gain about *A* from $B^A$ is defined as

$$I(B^A, A) = H(A) + H(B^A) - H(A, B^A) \quad \text{(V)}$$

$$= H(A) - H(A/B^A)$$

Suppose the traffic observed at *b* is proportional to *a* at any sampling period. If $B^A = j$, we can conclude that *A* equals to a fixed value *i*. In this case, we have $P(A = i|B^A = j) = 1$. This, according to *Eq. (III)*, makes the conditional entropy $H(A|B^A) = 0$. It means the uncertainty about the outcome of *A* when we know $B^A$ is 0. From *Eq. (V)*, we have $I(B^A,A) = H(A)$, implying that we gain the complete information about *A*, given $B^A$. Otherwise, if $B^A$ is independent of *A*, the conditional entropy $H(A|B^A)$ is maximized to *H(A)*. According to *Eq. (V)*, we have $I(B^A,A) = 0$, *i.e.*, we gain no information *A* from $B^A$. From *Eq. (V)*, we also figure out that we have to minimize the maximum mutual information $I(B^A,A)$ that any node can obtain about *A* to preserve privacy. In fact, since $B^A$ records the number of packets destined to node *a*, it can not be totally independent of random variable *A*. Therefore, the mutual information should be valued between the two extremes discussed above, *i.e.*, $0 < I(B^A,A) < H(A)$. This means that node *b* can still obtain partial information of *A's* traffic pattern.

Finally, we denote the average traffic through a node in a disjoint path as

$$T_{Avr} = \frac{1}{m}\sum_{i=1}^{m} T_i \quad \text{(VI)}$$

in which, *m* is the number of path found, $T_i$ is the traffic of a node at a specific time.

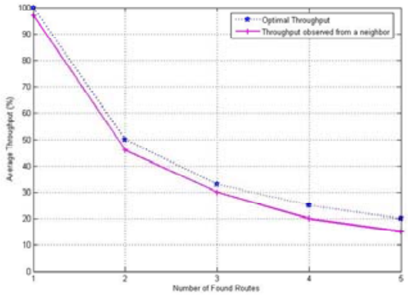


**Figure 6: Average Throughput Corresponding with Found Route**

We set up a simulation environment using *NS-2* and analyze the traffic of an intermediate node by the data observed from its neighbor. We analyze traffic in three cases regarding the number of found route (*m = 1, 2, 3*). The traffic is randomly distributed through found route in previous step, and at the same time, the total traffic simultaneously runs through those paths is 100 percent. It means the more number of found routes, the less data is transferred through a node, and the probability to capture the whole traffic is very small.

As shown in *fig. 6*, the obtained *Average Throughput* of a node in a route is always larger than the throughput observed by a neighbor of it. An intermediate node commonly can observe the data from its neighbor in wireless communication, but thanks for our proposed scheme, it can not know where is the data sent from and who is the receiver. Even if attackers can break key management scheme, the probability to compromise the whole intermediate nodes to capture all data is almost impossible.

In the *figure 7*, we monitor *Traffic Throughput* of a node by its neighbor in a period of time.

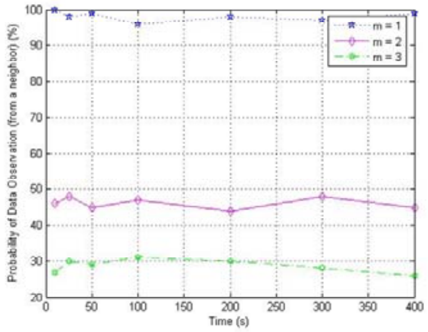


Figure 7: Traffic Observation Corresponding to Number of Found Routes

The figure has shown that the probability of successfully capture data will be reduced in direct proportion to the number of found routes (*m*). It means traffic privacy will be preserved in direct proportion to *m*.

## 6. Discussions and Conclusions

Our proposed approach in this paper is applied to WMNs which have static Mesh Router. In case of Wireless Mobile Ad-hoc Networks, it is much more difficult to maintain found routes according to the node's mobility. In fact, the routers which placed in a building are supposed to be physically protected. Therefore, they are harder to attack than the Transit Access Points (TAPs) which are placed outside. In this case, the source and destination nodes commonly are Access Points placed in buildings. Along with current key managements and authorization schemes, we assume that they are fully protected. If some attacks occur at intermediate nodes, as shown in previous sections, the probability that attackers can capture and restore data which is sent from source to destination through several disjoint paths is very small. Note that even if attackers can capture 99%, they still can not merge the data and this stolen data is meaningless.

After a route was found, the data is split and marked before it is sent to the destination. When other routes are found, the remaining packets will be continuously sent through those paths randomly. This mechanism will reduce time consumption and also preserve data confidentiality.

In our algorithm, we especially concern about reducing overhead, so that we propose two parameters as *Request_Time* and *Number_RREQ* (discussed in *section 3*) to avoid time consumption. Also, the algorithm is loop free thanks to the discarded *RREQ* and the finish of participating progress of unavailable nodes in *Step 2*.

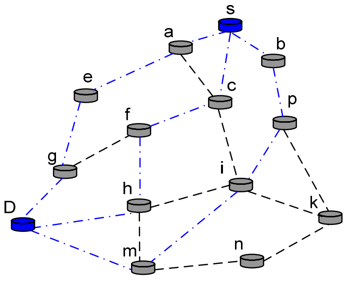


Figure 8: Example of 3-disjoint-path

The algorithm needs a small change in routing table and can be easily applied to the current routing platforms as discussed in *section 2*. Also, in our environment, there is enough number of nodes to find multiple disjoint paths. Of course, in the worst case, there is only one communication path (for example with only 3 mesh router) and this scenario becomes conventional communication (one route between source and destination).

In the future work, we will discuss attack scenarios and countermeasures regarding to security analysis and continue implementing our proposal in Testbed cooperating with existing routing protocol for WMNs. In addition, we will provide specific analysis how our scheme is implemented with well-known encryption algorithms to make the communication route more secure. Also, we are working on an algorithm for privacy preservation in Mobile Wireless PAN in which the network topology always changes due to node's mobility.

## References

[1] Cao Trong Hieu, Tran Thanh Dai, Choong Seon Hong, "Adaptive Algorithms to Enhance Routing and Security for Wireless PAN Mesh Networks", *OTM Workshops 2006*, *LNCS 4277*, pp. 585 – 594, 2006.

[2] R. Karrer, A. Sabharwal, and E. Knightly. "Enabling large-scale wireless broadband: The case for taps", *In HotNets*, 2003.

[3] V. Gambiroza, B. Sadeghi, and E. Knightly, "End-to-End Performance and Fairness in Multihop Wireless Backhaul Networks," *Proc. MobiCom*, 2004.

[4] Ben Salem, N.; Hubaux, J.-P., "Securing wireless mesh networks", *Wireless Communications, IEEE*, April 2006 Page(s):50 - 55

[5] M. Kodialam and T. Nandagopal, "Characterizing the Capacity Region in Multi-Radio Multi- Channel Wireless Mesh Networks" *Proc. MobiCom*, 2005.

[6] M. G. Reed, P. F. Syverson, and D. Goldschlag, "Anonymous connections and onion routing", *IEEE Journal on Selected Areas in Communications*, 16(4):482–494, 1998.

[7] Shu Jiang; Vaidya, N.H.; Wei Zhao, "Preventing traffic analysis in packet radio networks", *DARPA Information Survivability Conference & Exposition II*, 2001. DISCEX '01, Proceedings Volume 2,12-14 June 2001 Page(s):153 – 158.

[8] X. Wu and B. Bhargava, "Ao2p: Ad hoc on-demand position-based private routing protocol", *IEEE Transactions on Mobile Computing*, 4(4):335–348, 2005.

[9] David Goldschlag, Michael Reed, Paul Syverson. "Onion Routing for Anonymous and Private Internet Connections", *Communications of the ACM*, Volume 42 , Pages: 39 – 41, February 1999

[10] Klaus Becker, Uta Wille. "Communication Complexity of Group Key Distribution", *In 5th ACM Conference on Computer and Communications Security*, Pages: 1-6, 1998.

[11] Asokan, N., and Ginzboorg, P., "Key agreement in ad-hoc networks", *in Computer Communications*, vol. 23, p. 1627 – 1637, 2000.