Journal of Theoretical and Applied Information Technology
20th December 2015. Vol.82. No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

SECURE NEIGHBOR VERIFICATION PROTOCOL IN WIRELESS MESH NETWORKS

[1]P SUBHASH, [2]S RAMACHANDRAM

[1]Assoc Prof., Department of CSE, JITS, India

[2]Professor., Department of CSE, Osmania Universiy, India

E-mail: [1]subhash.parimalla@gmail.com, [2]schandram@osmania.ac.in

ABSTRACT

The main motivation of an attacker is to convince two far away nodes as neighbor nodes using wormhole attack easily without the knowledge of cryptographic primitives. Thus, it can significantly degrade the performance of Wireless Mesh Networks (WMNs). Secure neighbor discovery is a fundamental requirement of network nodes to ensure secure data communication. An adversary that bypass neighbor discovery process of a legitimate node using wormhole attack can disrupt the overlying protocols and applications. In this paper, we propose a secure neighbor verification mechanism to thwart wormhole attack that can prevent bogus links from being involved in the network operations. It employs node ranking scheme to compute relative distance between neighbors and uses connectivity information to check the genuinity of neighborhood creation. We evaluate our mechanism using simulation to demonstrate the efficiency in the presence of wormholes.

Keywords: *Secure Neighbor Verification;Wormhole Attack;Bogus Links;Relay Attacks;Wireless Mesh Network;Ranking Mechanism;Connectivity Information.*

1. INTRODUCTION

Wireless Mesh Network (WMN) has emerged as a new technology to support various application scenarios [1], such as broadband home networking, neighborhood and community networking, metropolitan area networking, etc. Unlike traditional wireless networks, each access point is connected to the fixed network, in WMN a subset of Mesh Routers are required to be connected to fixed network. As shown in Figure 1, a mesh router that is connected to the fixed network is called Mesh Portal Point(MPP), acts as a gateway router. A mesh Router that does not connect to the fixed network is called Mesh Point(MP). A mesh router connecting with mesh clients, provide access services to the mesh clients is known as Mesh Access Point(MAP). Many researchers have tried to improve the performance of wireless mesh networks, as it has been getting more attention that many applications are depending on them.

There are certain applications which require a high level of security, such as military applications. Due to the openness of the wireless medium, WMNs are susceptible to various kinds of internal and external threats. The issues related to various threats have been investigated in [2]. One such attack that causes severe impact on a wireless mesh network is a wormhole attack (or) Relay attack. A Wormhole is a low-latency link between two parts

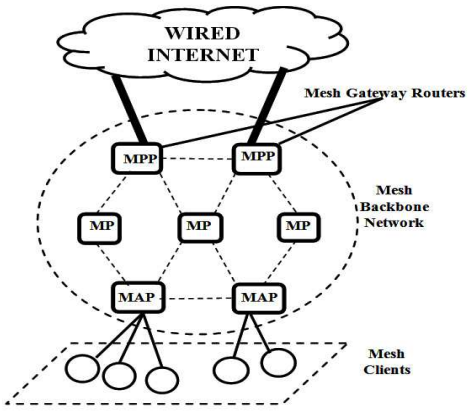

Figure 1: *Architecture of WMN*

of the network through which an attacker tunnels network messages from one point to another point using an out-of-band high-speed communication link or can employ in-band tunneling approach to

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

bypass intermediate nodes. This wormhole link is usually established between two colluding nodes located far away in the network. One of the main intentions of launching a wormhole attack by an attacker is to disrupt the neighbor discovery protocol. Neighbor discovery is the process by which a node determines the other nodes to which it can form a link in a single hop distance. An adversary Neighbor discovery protocol can show severe impact on routing protocols, and other overlying protocols. Thus, the design of secure neighbor discovery protocols over wireless mesh networks has proved to be a challenging task and the security enhancements of neighbor discovery protocols must provide the defense mechanism against wormhole attack to guard the network. In this paper, we employ a simple mechanism to secure neighbor discovery process by detecting the formation of false neighborhood information/fake links. The main objective of this work is to prevent wireless mesh networks from adversaries that launch relay attacks or wormhole attacks to disrupt the neighbor discovery process. The neighbor verification process is carried out next to the peer link establishment in wireless mesh network to check for true neighborhood creation. We propose a link verification mechanism that employs Check Request and Check Response frames to detect the malicious links that are being established during neighbor discovery process. We consider hop count value received in the check response packet during the link verification phase to mark a link status either valid or invalid. This verification process is applied between all pairs of nodes in the network. The preliminary version of this work was discussed in [3].

The rest of the paper is organized as follows: We present related work in Section 2. Section 3 describes network and threat model. In section 4, we present The proposed secure neighbor verification protocol for wireless mesh networks. In Section 5, we discuss security analysis. Section 6 describes the experimental study of the proposed protocol. In Section 7, we discuss in detail the implications of designed protocol. Finally, we conclude our paper in Section 8.

2. RELATED WORK

Many solutions to thwart against wormhole attack have been proposed in the literature and they are closely related to secure neighbor discovery. Most of the existing approaches to thwart wormhole attacks have been designed with the support of clock synchronization, an additional hardware, accurate time measurements, etc.

Hu et al. introduced a packet leashes technique [4] to detect wormholes in wireless networks, which may be either temporal leashes (or) geographical leashes. A leash is used to restrict the packet's maximum transmission distance and a leash is information which is added to a packet. Geographical leashes and temporal leashes both are used to defend against wormhole attack in the network. Since the receiver of a packet is able to detect if the packet traveled further than the leash allows. But this method requires GPS and tightly synchronized clocks.

3. The approach of Hu and Evans is a cooperative protocol [5] in which directional information is shared among nodes to prevent wormhole attack. However, This method does not require location information and clock synchronization, but requires all nodes in the network to be equipped with an additional hardware.

4. A scheme using transmission time based mechanism for the detection of wormhole attacks (TTM) was discussed by Van Tran and Xuan Hung in [6]. This method calculates every Round Trip Time (RTT) between two successive nodes along the route. Each node in the path will calculate RTT between it and the destination, this value will be sent back to the source node. Wormholes can be detected based on the RTT value as the RTT value between two fake neighbors is greater than the RTT value between two real neighbors.

5. A wormhole attack prevention algorithm (WAP) was proposed by Choi and Kim in [7]. In this approach, nodes monitor its neighbor's behavior when they send RREQ messages to the destination with the help of neighbor list. If the RREP message does not received by the source node within a stipulated time, it can detect the existence of wormhole attack. Once wormhole is

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

identified, source node records them in its wormhole node list. WAP is capable of detecting both the exposed and hidden attacks without the need of any specialized hardware. This approach does not fully support DSR as it is based on end-to-end signature authentication of routing packets.

DeWorm protocol [8] proposed by Thaier et.al. uses routing discrepancies between neighboring nodes along a path from a source to the destination to detect the presence of wormhole attacks in the network. This protocol is simple and localized. This method needs no extra hardware, synchronization (or) location and can able to detect physical layer wormholes.

WARP is a Wormhole Avoidance Routing Protocol [9], it considers link-disjoint multipath during path discovery, but eventually uses only one path for data transmission. WARP avoids wormhole attacks by anomaly detection and it is based on adhoc on-demand routing protocol (AODV) [10]. Every node in WARP maintains the anomaly values of its neighbors in its routing table. WARP enables the neighbors of the wormhole nodes to discover that the wormhole nodes have an abnormal path attraction.

Preventing Wormholes in Multi-hop Wireless Mesh Networks was discussed in [11], that focus wormhole attacks launched by an interior colluded malicious nodes referred to as a byzantine wormhole attack. This kind of attacks is more difficult to defend against, because they possess cryptographic primitives. The proposed mechanism relies on digital signatures and prevents formation of wormholes during the route discovery process using large discrepancy values in metric and hop count reported by various paths. It is designed for a mandatory path selection protocol in wireless mesh networks like hybrid wireless mesh protocol. This approach is simple, software based and does not require network nodes to be equipped with an extra hardware.

Matam Rakesh et. al [12] proposed WRSR (Wormhole Resistant Secure Routing) protocol that detects the existence of wormhole during route discovery phase and quarantines it. This protocol is based on neighborhood connectivity information and relies on existence of shorter alternate sub-paths to defend against byzantine wormhole attacks. WRSR uses Unit disk graph to determine necessary conditions to separate genuine path from wormhole path. WRSR addresses both hidden and exposed kind of attacks and it does not require any specialized hardware.

Poturalski et. al [13] proposed a formal investigation of possibility of secure neighbor discovery, which consider two general classes of protocols: time based (T-Protocols) and time and location based protocols(TL-Protocols), derive an impossibility result. Which also notify the conditions under which the impossibility result is lifted.

A scheme proposed in [14] is a secure neighbor verification protocol for constrained Wireless Sensor Networks, each node estimates the distance to its one hop reachable nodes. Then, nodes exchange their estimated information. Next, it detects topological distortions created by wormhole attacks by using a series of simple geometric tests performed by each node. The nodes that have successfully passed the tests are only verified to be genuine communication neighbors. This protocol is secure against the class 2- end wormhole attack model.

Hayajneh Thaier, et al. [15] proposed a scheme for secure neighborhood creation(SECUND) in wireless adhoc networks using hop count discrepancies. The main idea is to check links between every pair of nodes for the wormhole existence and to remove fake links without removing legal links. SECUND can able to detect and remove two-ended wormholes and multi-ended wormholes, which have not been addressed in DeWorm [8].

SEDINE [16] is an approach proposed for secure neighbor discovery through overhearing in static multi hop wireless networks, the main objective of this approach is to prevent a legitimate node from adding a non neighbor node as a neighbor to its neighbor list, in the absence of packet losses. SEDINE assume no out-of-band channel or power controlled transmission used by malicious nodes during the neighbor discovery protocol and it is proven to provide provable guarantees under a special class of attacker models. This method does not consider DOS attacks that prevent formation of true neighborhood links, physical destruction of nodes and physical layer jamming attacks.

Wu, Guowei, et al. [17] proposed a highly

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

efficient wormhole detection approach, that uses the local neighborhood information to calculate the transmission range. The neighbor list information is exchanged between neighbors through periodical beacon messages. Finally, detection of the wormhole is based on the transmission range that exploits the local neighborhood information check and it does not require specialized hardware or clock synchronizations.

Stoleru, Radu, et al. [18] detailed a mobile secure neighbor discovery (MSND) to guard the network against wormholes for secure neighbor discovery and wormhole localization in mobile adhoc networks. MSND allows neighbors to verify that they are communicating directly with each other. Detection of the wormhole is due to the fact that the path traveled by a ranging signal varies expected values when a wormhole exists. Instead of moving directly to the remote node, the ranging signal must move to one of the wormhole ends, transit the wormhole, and then exit to arrive at the destination node. MSND leverages graph rigidity concepts for the detection of wormholes.

3. NETWORK AND THREAT MODEL

This section presents the network an adversary model considered in the design of the proposed protocol.

3.1. Network Model

We consider a backbone of mesh routers that work co-operatively and route frames in a multi-hop fashion. The access services are provided by mesh routers with access point functionality (MAP). MRs (Mesh Routers) are wireless routers that adopt the enhanced distributed channel access (EDCA) mechanism as the basic MAC layer access mechanism. They implement the default hybrid wireless mesh protocol (HWMP) to relay multi-hop MAC data traffic. HWMP [19] combines the concurrent operation of proactive mode and an on-demand path selection mode derived from AODV Protocol [10]. Mesh gateways are mesh routers with gateway functionality and relay network traffic between other networks like the internet and WMN. The major aim of a WMN is to provide Internet connectivity as well as to support end-to-end communications for MCs (Mesh Clients) via multi-hop transmission over MRs.

In addition to that we consider the network to employ following protocols. Mesh routers employ the authenticated and mesh peering exchange protocol (AMPE) to establish secure peer links. Key generation and key management are the integral part of the AMPE process that facilitates mesh routers to generate pair-wise master keys, pair-wise transient keys and group transient keys. These keys are later used to prevent replay and forgery of transmitting frames. Specifically, group transient keys are used to protect check request and check response frames used by the proposed protocol.

3.2. Threat Model and Assumptions

We consider a threat model with the following attacking capabilities. An adversary can be either external or internal to the network. An external attacker can launch a wormhole attack by overhearing messages in the network and relaying them into the other end of the network. The relay channel can be an out-of-band channel or can be a sequence of malicious nodes operating in tandem to transmit a message at the far end of the network. An out-of-band channel symbolizes a network link that can be formed between malicious MRs by using a transmission channel that is not currently in use, or can be a wired link. An adversary can also compromise a MRs and launch a similar kind of attack, by becoming a legitimate part of the network. Compromised MRs can operate indepen-dently or work in collusion. Since compromised MRs are legitimate part of the network and possess the required security keys, they can bypass the existing security mechanisms and manipulate key information in frames. We assume that the mesh routers have no energy constraints, a pair of public/private key and public keys of all other mesh routers are assigned to each mesh router. All the links are assumed to be bi-directional.

4. THE PROPOSED SECURE NEIGHBOR VERIFICATION PROTOCOL

The proposed secure neighbor verification protocol relies on the relative distance from ROOT node (Rank) to initially establish peer links. Links that violate the definition of secure peer-link are

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

identified and removed from the set of neighbor links during the verification (validation) phase. Nodes employ a simple check request/response mechanism to validate links. Since, links that have been established due to malicious activity of compromised nodes (relay of peer-link open and peer-link confirm frames), are identified during the validation phase, it forms the core of the protocol.

Prior to neighbor discovery, each node A obtains a rank from its parent node, during the authentication process. The rank represents the relative distance from the ROOT node in WMN. Initial rank of a node is the incremented value of its parent node's rank. However, a node can update its rank whenever it receives a route announcement message (RANN) from the ROOT. The rank updating process is shown in Algorithm 1. Rank allows a node to differentiate between its neighbors and non-neighboring nodes. Any node can only associate with nodes that have a maximum rank difference of 1. This is justified in a tree (Graph) structure, nodes can have links one-level-up with their parents and one-level-down with its children, or at the same level. Initial comparison of ranks allows a node to check the level of nodes with which it establishes peer-links. For better understanding, the proposed protocol is presented in two distinct phases (i) Neighbor Discovery (ii) Link Verification.

Algorithm 1 Ranking: Node N on receiving Proactive RANN(Route Announcement)

```
1:if (Duplicate RANN) then
2:      Discard RANN
3: else if (Rank == 0) then
4:      Rank = Hop-Count in RANN
5: else if (Rank > Hop-Count in RANN) then
6:      Rank = Hop-Count
7: else if (Rank < Hop-Count in RANN) then
8:      Rank = Rank
9: end if
```

4.1 Neighbor Discovery

Neighbor discovery process begins when a node transmits an authenticated mesh peer-link open frame (hereby referred to as a HELLO message for simplicity) to all the nodes in its transmission range, including its rank. We assume that these HELLO messages are authenticated using the security mechanisms provided by authenticated and mesh peering exchange protocol presented in the standard. A node that receives this HELLO message, checks for its authenticity by verifying its signature. Later, it checks for the rank in HELLO message and accepts it only if the difference in rank is at most 1. Each node maintains a list of its neighbors and its neighbor's neighbors (i.e. Two-hop nodes), facilitated by exchanging neighbor lists. Once a node receives all the neighborhood information, it identifies the verifier nodes for its links. A verifier node for a link A↔B is a common node that share links with both A and B. After exchanging neighbor lists, each node identifies verifiers for its links and for links to which it is a verifier.

4.2 Link Verification Phase

After establishing tentative wireless links, a node initiates the verification process for each of its links to validate them. Each node independently carries out this link validation process. It begins when node A, the link verification initiator, broadcasts an authenticated check request towards node B, to verify the link A↔B. Link verification process is detailed in table 1. The check request contains a node's identity (node address), its rank, and the address of another node (B) of a link being verified. Nodes that identify themselves as verifiers for link A↔B, receive the check request and wait for a check response from B. Node B, the destined receiver, that receives this check request, verifies its validity and broadcasts a check response, after including the sender's rank (A's Rank), hop-count (set to 0), sequence number (SQ_NO) and its node ID. The significance of including senders rank in check response is detailed in the discussion section. Verifier nodes ({C}) that receive a valid check response (authentic) will re-broadcast the response message after updating the hop-count. Upon receiving the check response from at least one verifier, the initiator node (A) updates the link status as valid.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 1: Link Verification Process

Initiator of Link Verification Process (Node A)
1: Begin Process
2: Broadcast authenticated CheckRequest [A, B, Rank A]
*3: **If** ((Verifier && CheckResponse) is valid) **then** /* Executed when node A receives a valid CheckResponse */*
*4: **If** (Hop-Count == 1)*
Link-Status = Valid
*5: **Else***
Link-Status = In-Valid
6: End If
7: End Process
Verifier of Link (Node C)
1: Begin Process
*2: **If** (CheckResponse is valid && (Timer Active)) /* Executed by a Verifier C on receiving a CheckResponse */*
Label as duplicate CheckResponse
Suppress received CheckResponse
3: End If
4: Else
*5: **If** (CheckResponse is valid && (Timer In-Active))*
Update Hop-Count
Broadcast CheckResponse
6: End If
7: End Process
Other Nodes (Node X)
1: Begin Process
*2: **If** (Duplicate CheckResponse) /* Executed by any other node X on receiving a CheckResponse */*
Label as duplicate CheckResponse
Suppress received CheckResponse
3: End If
4: Else
*5: **If** (CheckResponse is valid && ($\lvert Rank_A - Rank_X \rvert \leq 2$))*
Update Hop-Count
Broadcast CheckResponse
6: End If
7: Else
Discard CheckResponse
8: End Else
9: End Process
Destined Receiver (Node B)
1: Begin Process
*2: **If** (CheckRequest is Valid) /* Executed by node B on receiving a CheckRequest */*
Create CheckResponse [A, B, Rank A, SQ_NO]
Broadcast CheckResponse
3: End If
4: Else
Discard CheckResponse
5: End Else
6: ***End Process***

On the other hand, after broadcasting the check response, each verifier node sets a timer whose value set to ($\alpha + \delta$), where α is the round trip time to it's farthest neighbor and delta (δ) is to capture small delays in the network and processing.

Other nodes ({X}) except verifiers, initiator and destined receiver that receive a valid check response, compute the rank difference between it's own rank and the rank received in the check response. If the absolute difference is greater than 2, such nodes simply ignore the check response. Otherwise, they update the hop-count and re-broadcast it. Other nodes also employ a suppression mechanism to avoid processing of duplicate check responses.

A verifier node that receives a check response within the set time interval (i.e $\alpha + \delta$), discards it. Timers facilitate a verifier to discard duplicate check responses that traverse up and down between same neighboring nodes. However, after the expiry of set time interval, the verifier nodes again accept check responses of the same session. This allows a verifier node to accept a check response that has traversed multiple hops when messages are relayed. This multiple processing of check responses highlights the point that only the verifier node can send a check response multiple times, to indicate both validity or invalidity of a link. In case of an invalid link, check the response sent by the other node B traverses towards node A across multiple hops, which is received by a verifier node and is accepted (as the timer would have expired) and sent to node A. On receiving such a check response node A invalidates the link A↔B

5. SECURITY ANALYSIS

Depends upon the ability to prevent nodes from establishing fake links, we note that our neighbor verification protocol, defend against wormhole attack in all of 2 cases and also discuss the detection operation when there is no wormhole.

***Case* 1**: *M_1 and M_2 are two colluding external malicious nodes*

Where malicious nodes tunnel the packets from one point of the network to another using an out-of-band channel to create bogus links. In this scenario, the external malicious nodes are placed in the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

network strategically by an adversary and they are not part of the network. Our solution prevents such bogus links from being involved in the network activity during the link verification process.

For example, Consider a wormhole link between two colluded external malicious nodes M_1 and M_2 in Figure 2, this attack can be launched by an adversary without possessing any security credentials in the network and relay the network messages from one point of location to another point via an out-of-band link established between M_1 and M_2. Nodes that are located in the communication range of M_1 believe that they are one hop neighbors to nodes that are located in the communication range of M_2. In effect, wormhole creates several bogus direct links between nodes of M_1's and M_2's area easily, even though they are originally located several hops away. In Figure 2, each node in M_1's range can form a direct bogus link with all the nodes in M_2's range and vice versa. Thus, it can form several bogus links. However, an attacker can limit the formation of a number of bogus links to normal range and can prevent malicious activity from being detected. The existing cryptographic methods cannot detect such bogus links as the malicious nodes M_1 and M_2 can simply relay the network messages without modifying them. As a result, a bogus link between node 2 and node 38 (2 ↔38) is established. This link is verified during the verification phase to mark the status as Invalid as node 2 and node 38 are located several hops away.

The verification process of link 2↔38 begins when node 2 broadcasts an authenticated check request towards node 38. The check request contains the address of node 2, it's rank, and the address of node 38. Node that identify themselves as verifier nodes that receive a valid check response (authentic) will re-broadcast the response message after updating the hop-count. Upon receiving the check response from at least one verifier, the initiator node (node 2) updates the link status valid. On the other hand, after broadcasting the check response, each verifier node sets a timer whose value set to $(\alpha + \delta)$, where is the round trip time to it's farthest neighbor and delta is to capture small delays in network and processing.

Other nodes except verifiers, initiator and destined receiver that receive a valid check response, compute the rank difference between it's own rank and the rank received in the check response. If the absolute difference is greater than 2, such nodes simply ignore the check response. Otherwise, they update the hop-count and re-broadcast it. Other nodes also employ a suppression mechanism to avoid processing of duplicate check responses.

A verifier node that receives a check response within the set time interval (i.e $\alpha + \delta$), discards it. Timers facilitate a verifier to discard duplicate check responses that traverse up and down between same neighboring nodes. However, after the expiry of set time interval, the verifier nodes again accept check responses of the same session. This allows a verifier node to accept a check response that has traversed multiple hops when messages are relayed. This multiple processing of check responses highlights the point that only the verifier node can send a check response multiple times, to indicate both validity or invalidity of a link. In case of an invalid link, the check response sent by the node 38 traverses towards node 2 across multiple hops, which is received by a verifier node(s) and is accepted (as the timer would have expired) and sent to node 2. On receiving such a check response node 2 invalidates the link 2↔38. The common neighbors of node 2 and node 38 are {1, 6, 35, 42}, acts as verifiers for the link.

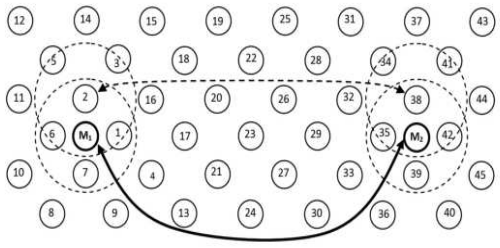

Figure 3: *A Network with External wormhole Attack*

Upon receiving a relayed check response packet, verifier nodes (Node 1 and Node 6) rebroadcast it after incrementing the hop count value and set the time interval (i.e $\alpha + \delta$). Node 2 receives the check response from any of the verifiers with hop count value 1 and set the link (2↔38) status to Valid. Later, verifier nodes (Node 1 (or) Node 6) receive the check response sent by the other node 38 traverses towards node 2 across multiple hops (as the timer would have expired) and sent to node 2. Upon receiving such check response having hop

Journal of Theoretical and Applied Information Technology
20th December 2015. Vol.82. No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

count value greater than one, node 2 set the link status to invalid and those invalid links cannot be considered further by the overlying protocols and applications. Thus, avoiding fake links from being involved in the network activities.

***Case* 2**: *Where, Node P and Node Q are two colluding internal malicious nodes*

Being a part of network, malicious nodes(Node P and Node Q) can bypass network messages from one point to another using an out-of band channel to create bogus links. Internal threats are much more complex to defend against than External threats because they possess legitimate keys. Our solution prevents such bogus links from being involved in the network activity.

Consider the scenario in Figure 3, where a link (A↔B) is established by an adversary. However, Node A and node B are located at different places in the network and they are convinced as neighbors. During link verification phase, node A broadcast check request packet towards node B to verify the link A↔B. In response to check request, node B broadcast check response packet. This verification process is similar to Case 1. Check response is relayed from node Q to node P and replayed at node P . Node P and node C are acting as verifiers for the link A↔B. Since, node P is compromised, the check response packet broadcasted by node C is more evident at node A to validate (or) invalidate the link. Initially the malicious link status is set to valid but later node A invalidate the link after receiving check response packet having traveled across many hops with hop count value greater than one via node C.

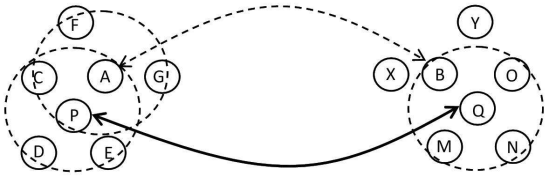

Figure 3: *A Network with Internal wormhole Attack*

***Case* 3:** *Detection operation when there is no presence of wormhole*

In this case, we show the detection operation of the protocol when there is no existence of wormholes. In Figure 4, a genuine link A↔B is established during neighbor discovery process, meaning that they are neighbors to each other. After establishing tentative wireless links, a node initiates the verification process for each of its links to validate them. Each node independently carries out this link verification process. Node A broadcast check request packet during the verification of a link A↔B.

The check request contains node A's address, its rank, and the address of other node (B) of a link being verified. Nodes that identify themselves as verifiers (node C) for the link A↔B, receive the check request and wait for a check response from B. Node B, the destined receiver, that receives this check request, verifies it's validity and broadcasts a check response, after including the sender's rank (A's Rank), hop-count(set to 0), sequence number (SQ_NO) and it's node ID.

Node A accepts check response packet only from the verifier nodes for the link (A↔B) being verified. Verifier nodes that receive a valid check response (authentic) will re-broadcast the response message after updating the hop-count. On the other hand, after broadcasting the check response, each verifier node sets a timer whose value set to (α + δ), where α is the round trip time to it's farthest neighbor and delta is to capture small delays in network and processing. A verifier node that receives a check response with in the set time interval (i.e α + δ), discards it. Timers facilitate a verifier to discard duplicate check responses that traverse up and down between same neighboring nodes

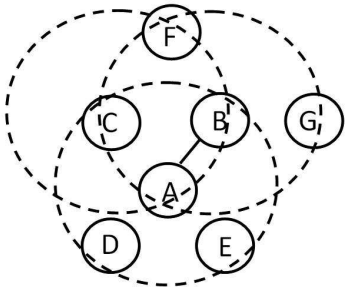

Figure 4: *A Network without wormhole Attack*

In this case, node C receives check response packet from node B, update the hop count value and rebroadcast it. Now the initiator of check request, node A receives the check response with hop count value 1, then node A set the link status as valid. This is how the genuine links of the network validate themselves.

Journal of Theoretical and Applied Information Technology
20th December 2015. Vol.82. No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

6. EXPERIMENTAL STUDY

In this section, we present the experimental results carried out to evaluate the performance of the proposed secure neighbor verification protocol (SNVP). The experiments were carried out in Omnetpp-4.2.1 discrete event network simulator [21]. We consider the 802.11s MAC protocol to perform the following experiments. At first, we evaluate the detection rate of SNVP. Detection rate acts as an important parameter for a detection protocol like SNVP.

We do not consider a wormhole link of length 2 due to the following reasons. When the length of the wormhole is 2 hops, a verifier node receives a relayed check response through malicious node, as well as a valid check response through a destined receiver before expiry of the timer. To recollect, each verifier node sets a timer whose value set to (α + δ). The aforementioned case happens because, a valid check response would be relayed by malicious node lying in the vicinity of verifier. As, an actual check response (traversed and not relayed) also reaches the verifier in 1 hop before the expiry of the timer, such a response is suppressed. Since, a wormhole of length 2 hops has negligible impact on the network, thus can be safely ignored.

Figure 5 presents the detection rate of SNVP for varying node degree (β) to detect malicious links in a network of 50 nodes. Initially the node degree (β =3) is considered. Lower node degree implies fewer alternate paths through which a check response can be received. However, when the node degree gradually increases the detection rate of SNVP increases rapidly. This can be attributed to the existence of paths through which a valid check response traverses in more than 2 hops, and reaches the verifier after the timer expires, thus allowing the verifier to accept and transmit towards the initiator. An initiator that receives a valid check response that has traversed more than 2 hops, marks the link as invalid

Next, we analyze the detection rate of SNVP by varying length of wormhole link for node degree of (β=4) in a network of 64 nodes (8X8 grid). Figure 6 summarizes the performance of the proposed solution. It gives an absolute 100% detection rate for a wormhole of length≥ 4. This is due to the fact that for any valid check response broadcasted by a

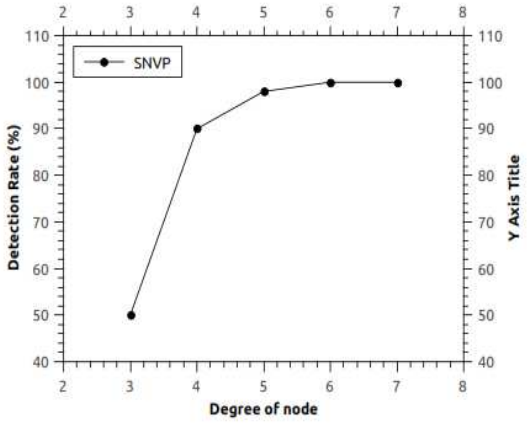

Figure 5: Detection Rate of SNVP for Varying Node Degree

destined receiver that is ≥4 hops away, would reach the verifier only after the expiry of its timer. Thus allowing the verifier to accept such a check response, and transmit it towards the initiator. An initiator invalidates this non-existent link after receiving such a check response. This remains true for all wormhole links of length ≥4.

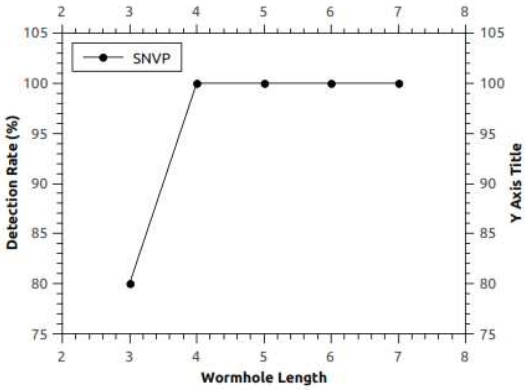

Figure 6: Detection Rate of SNVP for Varying Wormhole Length

6.1 False Positives in SNVP

False positive in SNVP is a situation where a valid link is shown as a wormhole (invalid link). This situation arises when a verifier node can be fooled to accept a check response that can potentially invalidate a valid link. This happens when a verifier node receives a check response after the expiry of its timer. Usually, a verifier node receives such a check response only when the destined receiver is more than 2 hops away. However, a compromised internal node can withhold the check response from a destined

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

receiver (which is the other end of the link, one-hop away) till the timer expires and later transmit it towards the verifier, forcing it to accept such a response. The impact of such an attack is restricted to its neighbors, which act as verifier for different links.

6.2 True Negatives in SNVP

True negative is another important parameter that characterizes a verification protocol like SNVP. It is a situation where a wormhole link goes undetected. The same experiment that presents the detection rate of fake links, also act as an indicator of true negatives. Table 2 presents the percentage of true negatives for a varying length of wormhole. As said, for a wormhole length of 3, true negatives arise in 50% of cases on average. Because, a check response packet traversed across multiple hops (sometimes it is 2 hops to reach a verifier in case of wormhole length 3) is discarded by the verifier node as the timer would have not been expired. But, for a wormhole length of ≥4, no wormhole link can pass as valid during the link verification phase.

Table 2: True Negatives in SNVP

Wormhole Length	3	4	5	6
True Negative(%)	50	0	0	0

6.3. Overhead analysis of SNVP

Finally, we analyze the performance of SNVP in terms of number of check response frames generated for varying number of nodes in the network. We compare the performance of SNVP operating with and without the ranking scheme. In the absence of ranking scheme, a check response traverses the entire network, whereas ranking scheme prevents the traversal of check responses beyond 5 levels. This experiment is just to showcase the advantage of using ranking scheme rather than focusing on the accuracy of detection mechanism. Figure 7 presents the overhead in terms of number of frames. The results are in accordance with the theoretical results presented in [20]. For an average node degree of 4, and a single hop link between initiator and destined receiver, the check response traverses 2 levels (up and down) excluding the level of destined receiver. However, in the absence of such restrictive mechanism, the check responses travel across the network. The results clearly quantify the advantage of employed ranking scheme in terms of overhead.

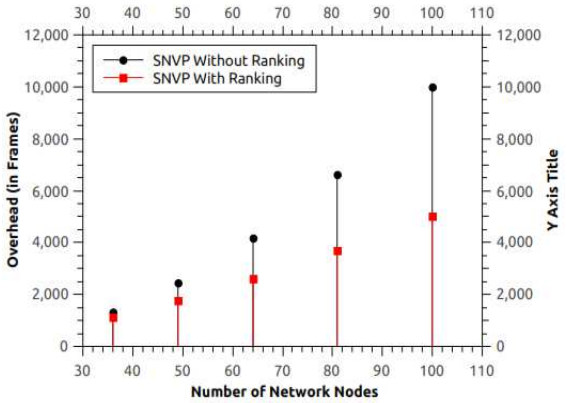

Figure 7: Overhead of SNVP with and without ranking Scheme

7. DISCUSSION

The proposed protocol uses the rank of a node while establishing peer-links. Gateway nodes that connect wireless mesh network to other networks like the internet are usually configured as root nodes. Root nodes enable the proposed protocol to ascertain nodes with ranks. Usage of rank restricts a node from accepting connections from arbitrary nodes. A node with rank Υ, can only accept connections with nodes having ranks Υ, ($\Upsilon\pm1$).

Ranks also restrict an attacker from targeting arbitrary nodes into forging non-existent neighbor links, since a node only accepts links that meet the aforementioned condition. Another important application of rank is during the transmission of check responses. A node transmits a check response only if the difference in ranks is at most 2. A value of 2 is chosen to accommodate the upper and lower bounds on a rank with respect to the initiator of the link verification process. Since, an initiator with rank Υ can form peer-links with nodes at level ($\Upsilon\pm1$), the neighboring nodes of a destined receiver can be at level ($\Upsilon\pm2$). Therefore, by restricting the transmission of check response to at most two levels (up or below), we aim to prevent unnecessary overhead in the network. On the other hand, the absence of ranking mechanism would result into significant overhead as each node that receives the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

check response re-broadcasts it.

Other aspect of the proposed protocol that plays a major role in the link verification process is the timing scheme. During the verification process, verifiers of a particular link maintain timers, and check for the sequence number in check responses, so as to distinguish between original and copies of a same check response. Check responses that are received when the timer is active are discarded. In case of a dubious wormhole link, the first copy of check response received by verifier nodes is considered valid, and transmitted towards the initiator. Other responses received before the expiry of verifier time interval are simply suppressed. Actual check response transmitted by the destined receiver is processed by verifier nodes, as it would be received after the expiration of the set timer. This second copy of the check response allows the initiator to invalidate a non-existent false link. It should be noted that the initiator accepts check responses only through a verifier node, which may be multiple.

In the absence of an attack, the initiator of the link verification process would receive a check response directly from the destined receiver. But, the link is considered to be valid only if the check response is received through the verifier. If the check response is not received due to any network conditions like collisions or interference, the initiator repeats the verification process in its entirety. The initiator repeats this process for a predefined (τ) number of times before the link is considered to be invalid.

SEDINE [16] is a similar kind of protocol that relies on k-verifiers that overhear and ascertain that a link is valid. To facilitate overhearing it assumes the absence of an out-of-band channel. This is an unrealistic assumption in case of a WMN, where nodes are deployed over a larger area and an attacker can easily establish an out-of-band link by simply tuning to frequency other than the operating one.

8. CONCLUSION

In this work, we propose a secure neighbor verification protocol to thwart wormhole attacks for the formation of true neighbor links between nodes in the network during neighbor discovery phase. It relies on a ranking mechanism to compute relative distance between neighbors, and employs connectivity information to validate those links using check request-response frames. The node ranking scheme is employed to limit check response frame from traversing to nodes at all levels. Thus, reducing the communication overhead and also restrict an attacker from targeting arbitrary nodes into forging non-existent neighbor links. The hop count value employed in a check response frame is used at the initiator node of check request to mark the status of a link being verified either valid (or) invalid.

REFRENCES:

[1] I.F.Akyildiz, X.Wang, and W.Wang, "Wireless mesh networks: a survey," *Computer networks*, vol. 47, no. 4, 2005, pp. 445–487.

[2] W. Zhang, Z. Wang, S. K. Das, and M. Hassan, "Security issues in wireless mesh networks," in *Wireless Mesh Networks. Springer*, 2008, pp. 309–330.

[3] P.Subhash and S.Ramachandram, "Secure neighbour discovery in wireless mesh networks using connectivity information," in *Advances in Computing, Communications and Informatics (ICACCI), 2015 Inter-national Conference on. IEEE*, 2015, pp. 2061–2066.

[4] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Packet leashes: a defense against wormhole attacks in wireless networks," in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, vol. 3. IEEE, 2003, pp. 1976–1986.

[5] L. Hu and D. Evans, "Using directional antennas to prevent wormhole attacks." in *NDSS*, 2004.

[6] P. Van Tran, Y.-K. L. Le Xuan Hung, S. Lee, and H. Lee, "Ttm: An efficient mechanism to detect wormhole attacks in wireless ad-hoc networks," in *Consumer Communications and Networking Conference*, 2007. CCNC 2007. 4th IEEE, 2007, pp. 593–598.

[7] S. Choi, D.-y. Kim, D.-h. Lee, and J.-i. Jung, "Wap: Wormhole attack prevention algorithm in mobile ad hoc networks," in *Sensor Networks, Ubiquitous and Trustworthy Computing,* 2008. SUTC'08. IEEE Interna-

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

tional Conference on. IEEE, 2008, pp. 343–348.

[8] T. Hayajneh, P. Krishnamurthy, and D. Tipper, "Deworm: a simple protocol to detect warmhole attacks in wireless ad hoc networks" *in Network and system security, 2009.Nss'09, Third international conference* on IEEE,2009,pp.77-80

[9] M.-Y. Su, "Warp: A wormhole-avoidance routing protocol by anomaly detection in mobile ad hoc networks," *computers & security*, vol. 29, no. 2.

[10] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *Mobile Computing Systems and Applications,* 1999. Proceedings. WMCSA'99. Second IEEE Workshop on. IEEE, 1999, pp. 90–100.

[11] P. Subhash and S. Ramachandram, "Preventing wormholes in multi-hop wireless mesh networks," in *Advanced Computing and Communication Technologies (ACCT), 2013 Third International Conference on. IEEE*, 2013, pp. 293–300.

[12] R. Matam and S. Tripathy, "Wrsr: wormhole-resistant secure routing for wireless mesh networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2013, no. 1, pp. 1–12.

[13] M. Poturalski, P. Papadimitratos, and J.-P. Hubaux, "Secure neighbor discovery in wireless networks: formal investigation of possibility," in *Proceedings of the 2008 ACM symposium on Information, computer and communications security. ACM*, 2008, pp. 189–200.

[14] R. Shokri, M. Poturalski, G. Ravot, P. Papadimitratos, and J.-P. Hubaux, "A practical secure neighbor verification protocol for wireless sensor networks," in *Proceedings of the second ACM conference on Wireless network security. ACM*, 2009, pp. 193–200.

[15] T. Hayajneh, P. Krishnamurthy, D. Tipper, and A. Le, "Secure neighborhood creation in wireless ad hoc networks using hop count discrepancies," *Mobile Networks and Applications*, 2012, vol. 17, no. 3, pp. 415–430.

[16] S.Hariharan, N.B.Shroff, and S.Bagchi, "Secure neighbor discovery through overhearing in static multi-hop wireless networks," *Computer Networks*, 2011, vol. 55, no. 6, pp. 1229–1241.

[17] G. Wu, X. Chen, L. Yao, Y. Lee, and K. Yim, "An efficient wormhole attack detection method in wireless sensor networks," *Computer Science and Information Systems*, 2014, no. 00, pp. 68–68.

[18] R. Stoleru, H. Wu, and H. Chenji, "Secure neighbor discovery and wormhole localization in mobile ad hoc networks," *Ad Hoc Networks*, , 2012 vol. 10, no. 7, pp. 1179–1190.

[19] M. Bahr, "Update on the hybrid wireless mesh protocol of ieee 802.11s," 2007.

[20] Jacquet and L. Viennot, "Overhead in mobile ad-hoc network proto-cols," 2000.

[21] The OMNeT++ Network Simulator: http://www.omnetpp.org. Accessed September 2011.