TOPOLOGICAL COMPARISON-BASED WORMHOLE DETECTION FOR MANET

King Sun Chan[1], Mohammad Rafiqul Alan[2]
Department of Electrical & Computer Engineering
Curtin University, Bentley, Perth, 6102 Australia
k.chan@curtin.edu.au[1],m.alam5@postgrad.curtin.edfu.au[2]

ABSTRACT

Wormhole attack is considered one of the most threatening security attacks for mobile ad hoc networks (MANET). In a wormhole attack, a tunnel is setup in advance between two colluders. The colluders record packets at one location and forward them through the tunnel to another location in the network. Depending on whether or not the colluders are participating in the network functions, the wormhole attack can be further divided into two categories: traditional wormhole attack and Byzantine wormhole attack. Existing researches focusing on detecting traditional wormhole attacks can be classified into three categories: one-hop delay based approach; topological analysis based or special hardware/middleware based approaches. Unfortunately, they all have their own limitations. Most of the researches detecting Byzantine wormhole attack are not addressing the Byzantine wormhole attack directly. Instead, they focus on observing the consequence after a Byzantine wormhole attack, like packet dropping or modification. In this paper, we propose to detect both traditional and Byzantine wormhole attacks by detecting some topological anomalies introduced by wormhole tunnels. Simulation results show that our scheme can achieve both high wormhole attack detection rate and accuracy. Our scheme is also simple to implement.

KEY WORDS

Wormhole attack, Byzantine wormhole attack, manet, topological comparison.

1. Introduction

Mobile ad hoc network (MANET) attracted a lot of attention recently in networking community. Most of the previous ad hoc networking research has focused on routing protocols and communication methods in a trusted environment. However, many applications need secured communication [1]. MANETs are vulnerable to a number of security attacks due to their flexibility in network configuration, openness of the wireless medium and absence of any centralized controller. Recently, there are many papers focusing on providing security for MANETs. Authors in [2] proposed a password-authenticated group key exchange protocol for MANETs. Authors in [3] proposed two-layer INS concept to secure routing protocols. A detection framework called separation of detection authority is proposed in [4] for detecting selfish nodes on

MANETs. In this paper, we focus on one particular network layer attack: wormhole attack. Depending on whether or not the attackers are also actively participating in network layer functions, the wormhole attack can be further divided into two categories: traditional wormhole attack and Byzantine wormhole attack. In the traditional wormhole attack, an attacker overhears the packets in its vicinity, records them and then tunnels them to another location where they are replayed by another colluding attacker. As a result, two far away nodes consider themselves as one-hop neighbours. In the Byzantine wormhole attack, the attackers also participate in the network functions, like routing, flooding, authentication, etc. In Byzantine wormhole attack, when one colluder receives a packet, it first takes action according to the network function requirement, and then tunnels the packet to the other side of the tunnel. The other colluder, after receiving the forwarded packet from the tunnel, will then process it and then take appropriate action as if it is received from a direct legitimate neighbour. As the colluders also participate in all network functions, Byzantine wormhole attack is more difficult to detect. The motivation of this paper is to develop a unified scheme to detect both traditional and Byzantine wormhole attacks with high detection rate and simple implementation.

As short wormhole links may not attract a lot of traffic and will not be of much use to the adversary [5], we consider the wormhole tunnel to be at least two hops long in this paper. As a wormhole tunnel will introduce some topological anomaly, we focus on topological analysis to detect such topological anomaly for detecting both traditional and Byzantine wormhole attacks. Simulation results show that our scheme can achieve both high wormhole detection rate and detection accuracy.

The remainder of this paper is organized as follows. In Section 2, we review existing wormhole detection methods and their limitations. Our proposed solution for detecting traditional wormhole attack is presented in section 3. We extend our scheme to detect Byzantine wormhole attack in section 4. Finally, our conclusion is drawn in Section 5.

2. Related Works

Wormhole attack attracts a lot of attention in MANET security research community recently. In this section, we first briefly review existing schemes for detecting traditional wormhole attack. Then we move to cover the existing schemes for defending against Byzantine wormhole attack.

2.1 Existing methods for detecting traditional wormhole attack

Existing approaches for detecting traditional wormhole attack can be classified into three categories: one-hop delay based approach; topological analysis based or special hardware/middleware based solutions.

Schemes proposed in [6-12] are one-hop delay based. In [6], a wormhole detection method based on round trip time (RTT) and neighbour number is presented. When the RTT between two nodes is considerably longer, they check the neighbour number. If the value of neighbor number is greater than the average neighbor number, there is a suspect that a wormhole link is in between. This method assumes that all network nodes use the same hardware and software configuration. Moreover, they assume the network nodes are uniformed distributed, which may not be true in some mobile ad hoc networks. Schemes in [7-12] also rely heavily on measuring one-hop delays to detect wormhole attack.

Another approach of combating wormhole attacks is to use graph analysis. Maheshwari et al. in [5] proposed a wormhole detection algorithm which looks for forbidden substructure in the connectivity graph that should not be present in a legal connectivity graph. Unfortunately, this approach is very complicated and impractical to real system.Lee et al. in [13] propose a method where each node gathers information of its neighbors within two hops. Each newly joined node broadcasts an announcement, which is valid only within the next two hops. The requirement of maintaining two hops neighbors, keyed hash and TTL limit the applicability of this method in a distributed system where exists a wide variety of participants. Dong et al. in [14] propose to analyse the topological impact introduced by traditional wormhole tunnels. Unfortunately, the presented detection scheme requires the network to run Dijkstra shortest path routing algorithm which may be a heavy burden to many mobile ad hoc networks.

The solutions belonging to the third category of combating wormhole attack use a special hardware device, strict time synchronization or special network protocol. Packet leashes are used in [1] to detect and defend against wormhole attacks. However, the accuracy of GPS devices is low in presence of physical obstacles. Another detection method in [15] uses directional antennae to obtain relative directional information and verify possible neighbors. This method suffers from antenna's directional errors. In NEVO [16], a firmware up-gradation of the MAC layer is needed so that the sender can passively monitor the forwarding of broadcast type packets by its neighbors. Moreover, NEVO uses network layer verification, which is a time consuming task.

It can be seen that most of the existing traditional wormhole detection methods rely on measuring one hop delays. The major advantage of this type of solutions is simplicity and easy implementation. However, delays are not only caused by the presence of wormholes but also some other factors like link congestion, queuing delays, difference in intra-nodal processing capabilities etc. In our scheme, instead of only looking at the round trip delays, we turn to detect the topological anomalies introduced by the wormholes. Using this topological feature, we can

detect traditional wormhole tunnel with high accuracy.

2.2 Existing methods for detecting Byzantine wormhole attack

Most of the existing solutions for detecting Byzantine wormhole attack rely on encryption and authentication. As the compromised nodes will drop or modify the user packets, these solutions try to detect the dropped packets or modified packets to identify the Byzantine attacks. In ODBSR [17], reliability is chosen as the metric in routing. Each link and then path consisting of multiple links is assigned a weight. If the packet loss rate is over some threshold, the source launches the binary search to determine which link is in fault. The faulty link will be assigned a larger weight and eventually avoided in the future connection setup phase. In SRAC [18], the authors proposed an algorithm to detect Byzantine attacks by using both message and route redundancy during route discovery. Multiple copies of the same packet are received by a node. After comparing all multiple copies, it is possible to detect any missed or modified copies and identify the compromised nodes. Unfortunately, all these schemes are very complicated and only focusing on packet dropping or modification. If the compromised nodes are only interested in analysing traffic or spoofing, all these schemes fail. In our approach, we intend to address byzantine wormhole attack directly. We focus on detecting abnormal topological features introduced by byzantine wormhole tunnels. Therefore, it is possible that paths including Byzantine wormhole tunnels can be completely avoided and thus minimizing the adverse impact.

3. Detecting traditional wormhole attack

In this section, we present our proposed scheme for detecting traditional wormhole attack. This scheme will be extended to cover the Byzantine wormhole detection in section IV. Our scheme is based on the following two observations of traditional wormhole attacks:

1) Temporal anomaly: Two fake one-hop neighbours with a wormhole tunnel in between have longer RTT, compared to the RTT between two true one-hop neighbours.

2) Topological anomaly: Two true one-hop neighbours usually share common true one-hop neighbours among them, and two fake one-hop neighbours do not share common true one-hop neighbours.

3.1. Neighbor List Construction

Each node in the network maintains its own one-hop neighbor list and average RTT (RTT_{avg}) to its direct neighbors. The neighbor list consists of two parts: trusted and suspected. The nodes included in the trusted part are considered true direct neighbors while the nodes included in the suspected part are suspected as under the traditional wormhole attack. Nodes exchange HELLO and HELLO_REPLY packets for populating their neighbor lists. The HELLO packet contains the following fields: source node ID and sequence number. The

HELLO_REPLY packet contains the following fields: source field, destination field and sequence number. The details of the HELLO and HELLO_REPLY exchange are described in the following:

1. The source node broadcasts a HELLO packet to its one-hop neighbors.
2. A node receiving a HELLO packet unicasts a HELLO_REPLY packet back to the source node.
3. After receiving a HELLO_REPLY packet from one neighbor, the RTT between the source node and the neighbor is first measured. Based on the relationship between the measured RTT and the maintained RTT_{avg}, this neighboring node may be put into the suspected or trusted part of the node's neighbor list: if the measured RTT is greater than three times of the current RTT_{avg} ($RTT > 3 \times RTT_{avg}$), this neighboring node is included into the suspected part of the neighbor list; otherwise, this node is considered a trusted neighbor and $RTT_{avg}$is updated.

3.2 Calculating RTT_{avg}

Round Trip Time (RTT) is measured as the delay between when a HELLO packet is broadcasted and when the corresponding *HELLO_REPLY* is received by the initiator.

Each node maintains an average value of one-hop *RTT* between itself and its one-hop neighbors. This value, denoted by RTT_{avg} in this paper, is measured using the following formulae:

$$RTT_{avg(0)} = RTT_0$$

$$RTT_{avg(i)} = \left(RTT_{avg(i-1)} \times \propto\right) + \left((1-\propto) \times RTT_i\right)$$

3.3 SUS and *TRST* parts of a *Neighbor List*

Based on measured RTT, we can populate the SUS and TRST parts of a node's neighbor list. However, the list may not be accurate enough as RTT is affected by many factors. We use topological comparison to eliminate as many as possible true neighbors from the SUSP part to improve the detection accuracy.

3.4 Topological Comparison

As we have noticed in observation 2), if two nodes are true one-hop neighbors, they usually share other common true neighbors. However, the nodes around a traditional wormhole tunnel get a distorted view of the network topology. Therefore, two far away nodes consider themselves as direct neighbors but this particular topological feature may not be held anymore. We can use this property to improve the performance of our scheme.

If a node's SUSP part of the neighbor list is not empty, it sends ***ENQ*** packets to all nodes in its ***SUSP*** part of the ***Neighbor List***. In response to ***ENQ***, the recipients reply with their respective ***TRUS*** part of their ***Neighbor List*** back to the ***ENQ*** source. After receiving the reply, the node can use Algorithm 1 to conduct topological comparison. The parameters used are shown in Table I.

Table I parameters for Algorithm 1

S	Sender of *ENQ* packets
r	Receiver of *ENQ* packets
$TRUS_s$	*TRUS* list of *s*
$TRUS_r$	*TRUS* list of *r*
me	*TRUE/FALSE,* denotes whether s is in $TRST_r$
trusted	No. of nodes in $TRUS_r \cap TRUS_s$

Algorithm 1 Detectingtraditional Wormhole

1. **if***me* = *TRUE* **then**
2. Delete ***r*** from $\boldsymbol{SUSP_s}$
3. Insert *r* into $\boldsymbol{TRUS_s}$
4. **end if**
5. **else if** *me* = *FALSE* **then**
6. **if** (*trusted=0*) **then**
7. link with ***r*** contains wormhole tunnel
8. **else**
9. delete ***r*** from $\boldsymbol{SUSP_s}$
10. Insert *r* into $\boldsymbol{TRUS_s}$
11. **end if**
12. **end if**

3.5 Performance evaluation

We use ns-2 to evaluate our scheme. And parameters are listed in Table II.

Table II Simulation parameters

Simulation area	1000m×1000m to 1400m×1400m
Number of nodes	10, 15, 20, or 30
Transmission range	250m
Routing algorithm	AODV
Simulation runs per scenario	100

We use detection rate and detection accuracy as performance measurement of our scheme. Figure 1 shows the detection rate versus tunnel length for different network sizes. It can be seen that the detection rate increases as the tunnel length increases. It is because that, with longer tunnel length, the RTT between a pair of fake direct neighbours is longer and easier to be identified. It can also be seen that higher detection rate is achieved with larger network size. It is due to the fact that, with larger network size, each node has more genuine neighbouring nodes which leads to more accurate RTT_{avg} estimation and thus less likely to misclassify fake neighbouring nodes as trusted neighbours. We also compared the detection rate of our scheme with that of scheme developed in [7] in Figure 2. It can be seen that our scheme achieves much better performance. It can also be noted that the improvement of our scheme is more significant when the tunnel length is short. It is because that the scheme in [7] relies on RTT only to detect wormhole attack.

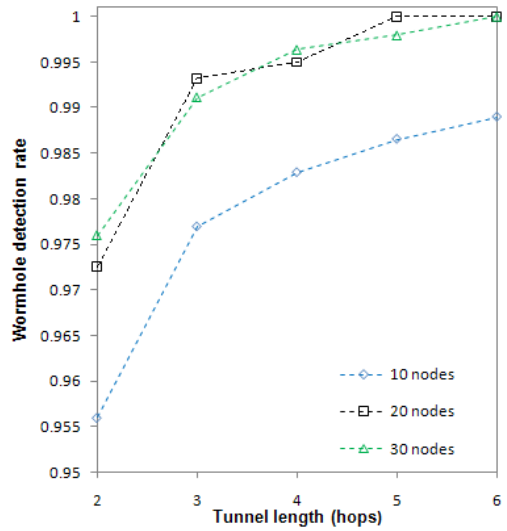

Figure 1 Detection rate vs. Tunnel length

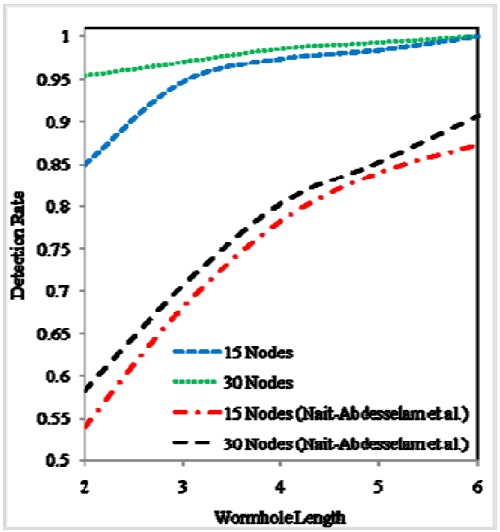

Figure 2comparisonwith the scheme in [7]

4. Detecting Byzantine wormhole attack

In this section, we turn our attention to detect the Byzantine wormhole attack. As we have explained earlier, two nodes with a Byzantine wormhole tunnel in between consider themselves as three-hop neighbors. However, as in the traditional wormhole attack case, a Byzantine wormhole tunnel will also introduce some anomalies which can be observed for detecting such attacks. Our scheme is based on the following two observations:

1) Temporal anomaly: two fake three-hop neighbors with a Byzantine wormhole tunnel in between have longer RTT.

2) Topological anomaly: two true three-hop neighbors have some of the other's one-hop neighbors as no-more-than-three-hop true neighbors, except the one involved in this pair's three-hop connection.

4.1 Neighbor List

Each node in the network maintains its own neighbor list which includes all neighbors no less than three hops away. The neighbor list consists of two parts: trusted part (TRUS) and suspected part (SUSP). As we only consider the Byzantine wormhole attack, we assume all one-hop and two-hop neighbors are true neighbors. Therefore, the suspected part only includes the three-hop neighbors which are considered under the Byzantine wormhole attack. The procedure for constructing the neighbor list is similar to Section 3 and we exclude the details due to space limit.

4.2 **Topological Comparison**

If two nodes are true three-hop neighbors, they usually consider some of the other's one-hop neighbors as trusted neighbors, except the one

involved in the pair's three-hop connection. However, the nodes around a Byzantine wormhole tunnel get a distorted view of the network topology. Therefore, two far away nodes consider themselves as three-hop neighbors but this particular topological feature may not be held anymore. Therefore, we can use this feature to detect Byzantine wormhole attacks in MANET. We modify the topological comparison algorithm we developed in section 3 for detecting Byzantine wormhole attack.

Table III Notations used in Algorithm 2

S	Sender of *ENQ* packets
r	Receiver of *ENQ* packets
$SUSP_s$	*SUSP* list of the neighbor list of s
ONL_s	*One-hop neighbor list of s*
$TRUS_r$	*TRUS* list of the neighbor list of r
ONL_r	*One-hop neighbor list of r*
me	*TRUE/FALSE,* denotes whether s is in $TRUS_r$
$trusted_s$	No. of nodes in $TRUS_r \cap ONL_s$
$trusted_r$	No. of nodes in $TRUS_s \cap ONL_r$

Each node in the network has its own ***Neighbor List***. After the neighbor discovery process a node sends ***ENQ*** packets to all nodes in its ***SUSP*** list of the ***Neighbor List***. In response to ***ENQ***, the recipients reply with their respective ***TRUS*** part of their ***Neighbor List*** back to the ***ENQ*** source. After receiving the reply, the node runs Algorithm 2 for detecting the Byzantine attack. The parameters used in comparison are shown in Table III. In this phase, the ***TRUS***parts of a ***Neighbor List*** is modified when a suspected node proves its credibility.

Algorithm 2 Detecting Byzantine Wormhole

1. **if***me = TRUE* **then**
2. Delete ***r*** from $\boldsymbol{SUSP_s}$
3. Insert *r* into $\boldsymbol{TRUS_s}$
4. **end if**
5. **else if** *me = FALSE* **then**
6. **if** ($trusted_s$*=1 and* $trusted_r$*=1)* **then**
7. connection with ***r*** contains Byzantine wormhole tunnel
8. **else**
9. delete ***r*** from $\boldsymbol{SUS_s}$
10. Insert *r* into $\boldsymbol{TRST_s}$
11. **end if**
12. **end if**

4.3 Simulation results

We have also simulated our scheme with simulator ns-2 to evaluate the performance. Simulation parameters are similar to those shown in table II, except that the simulated area is limited in a square field of size 1000*m* * 1000*m* with 30 randomly generated nodes.

Figure 3 shows the detection rate versus Byzantine wormhole tunnel length. It can be seen that detection rate increases as tunnel length increases. It is because the RTT between a pair of fake three-hop neighbours is greater with longer tunnel length, and therefore easier to be identified. In Figure 4, the

detection accuracy versus tunnel length is shown. It can be seen that our scheme can achieve very high accuracy.

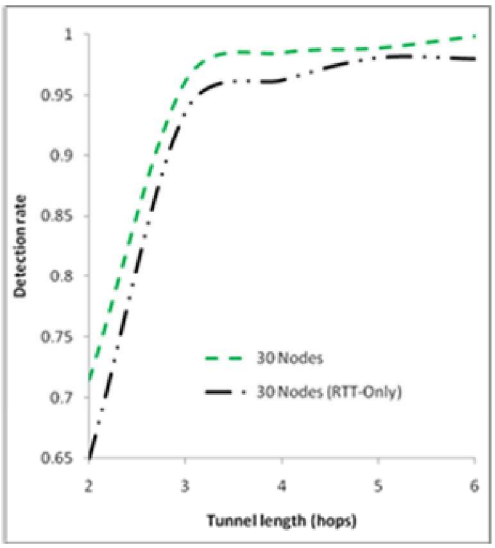

Figure 3Detection rate vs tunnel length

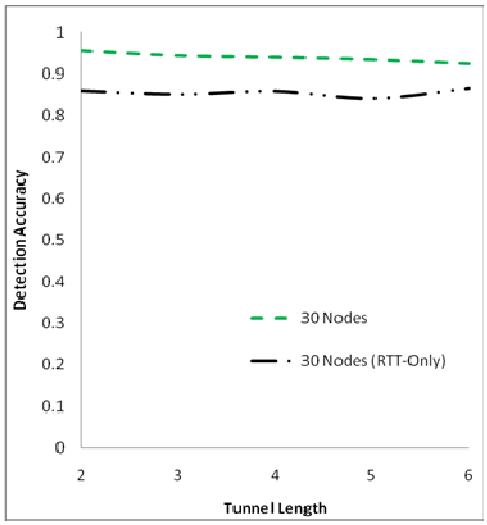

Figure 4Accuracy vs tunnel length

4. Conclusion

Wormhole attack is considered one of the most challenging and threatening security attacks in mobile ad ho networks. Most of the existing wormhole detection schemes focus only on traditional wormhole attacks. And they rely on observing longer RTTs between neighbouring nodes under the traditional wormhole attack which may lead to poor detection performance. Existing schemes for Byzantine wormhole attacks focus on the consequences of Byzantine wormhole attacks, like packet dropping and modification to detect the existence of Byzantine wormhole attacks. In this paper, we try to detect both traditional and Byzantine wormhole attacks directly. We propose to detect the topological abnormality introduced by the traditional and Byzantine wormhole attacks. By detecting wormhole attacks directly, those links under wormhole attack can be avoided completely during the routing phase and thus limiting the adverse consequence from wormhole attack to the minimum. Simulation results show that our scheme can achieve both high detection rate and accuracy of alarms. The implementation of our scheme is also simple.

REFERENCES

[1] Y.-C. Hu, A. Perrig, and D. B. Johnson, “Wormhole attacks in wireless networks,” IEEE JSAC, pp. 370–380, Feb. 2006.

[2] D. He, C. Chen, M Ma, S. Chan, and J. Bu, “A secure and efficient password-authenticated group key exchange protocol for mobile ad hoc networks”, International Journal of Communication systems, 2011.

[3] A. Le, J. Loo, A. Lasebae, M. Aiash, and Y. Lou, “6LoWPAN: a study on QoS security threats and countermeasures using intrusion detection system approach”, International Journal of Communication systems, 2012.

[4] Z. Chong, S. Tan, B. Goi, and B. Ng, ‘Outwitting smart selfish nodes in wireless

mesh networks", International Journal of Communication systems, 2011.

[5] R. Maheshwari, J. Gao, and S. R. Das, "Detecting wormhole attacks in wireless networks using connectivity information," in Proc. INFOCOM 2007, pp. 107–115.

[6] Z. Tun and A. H. Maw, "Wormhole attack detection in wireless sensor networks," in Proceedings of World Academy of Science, Engineering and Technology, 2008.

[7] F. Nait-Abdesselam, "Detecting and avoiding wormhole attacks inwireless ad hoc networks," IEEE Comm. Mag., pp.127–133, Apr. 2008.

[8] T. Van Phuong, N. T. Canh, Y.-K. Lee, S. Lee, and H. Lee, "Transmission time-based mechanism to detect wormhole attacks," in Proc. 2nd IEEE Asia-Pacific Service Computing Conference 2007, pp. 172–178.

[9] M. Khabbazian, H. Mercier and V. K. Bhargava, "Severity analysis and countermeasure for the wormhole attack in wireless ad hoc networks", IEEE trans. Wireless comm., pp. 736 – 745, 2009.

[10] S. Choi, D. young Kim, D. hyeon Lee, and J. il Jung, "Wap: Wormhole attack prevention algorithm in mobile ad hoc networks," in Proc. IEEE SUTC '08, 2008.

[11] H. S. Chiu and K. S. Lui, "DelPHI: wormhole detection mechanism for ad hoc wireless networks", Proc. Int'l Symp. Wireless Pervasive Comp., 2006.

[12] J. Biswas, A. Gupta, and D. Singh, "WADP: A wormhole attack detection and prevention technique in MANET using modified AODV routing protocol", Proc. 9th Int'l Conference on Industrial and Information Systems (ICIIS), 2014.

[13] G. Lee, J. Seo, and D. Kim. "An Approach to Mitigate Wormhole Attack in Wireless Ad Hoc Networks", in Proc. ISA2008.

[14] D. Dong, Mo Li, Y. Liu, X. Li, and X. Liao, "Topological detection on wormholes in wireless ad hoc and senser networks", IEEE/ACM trans. Networking, 2011, 19(6): p. 1787-1796.

[15] L. Hu and D. Evans, "Using directional antennas to prevent wormhole attacks," Proc. 11th Network and Distrbuted System Security Symposium, 2003.

[16] X. Su and R. V. Boppana, "Mitigating wormhole attacks using passive monitoring in mobile ad hoc networks," Proc. IEEE Globecom2008, 2008.

[17] B. Awerbuch, R. Curtmola, D. Holmer, *et. al.*, "ODSBR: An on-demand secure Byzantine resilient routing protocol for wireless ad hoc networks", ACM Trans. Inf. Syst. Secur., 2008. 10(4): p. 1-35.

[18] M. Yu, M.C. Zhou, and W. Su, "A secure routing protocol against byzantine attacks for MANETs in adversarial environments", IEEE Transactions on Vehicular Technology, 2009. 58(1): p. 449-460.