DE GRUYTER OPEN

BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **14**, No 2

Sofia • 2014

Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2014-0016

Secure Routing in Manet Using Synchronization and Data Authencity

Gundala Swathi, R. Saravanan

School of Information Technology and Engineering
Vit University, Vellore-632014, Tamilnadu, India
E-mail: gundalaswathi@vit.ac.in

Abstract: *In recent years synchronization plays a major issue for secure transmission in mobile adhoc networks. When an attacker modifies the time synchronization algorithm, the nodes will have faulty estimates of other nodes location, leading to chaos. While transmitting under these adverse conditions, packets might be lost or might be sent to wrong locations. Data replication and data diffusion are two methods which are used to solve the problem of data availability. In this paper we propose an algorithm for secure multi hop transmission used for external attacks.*

Keywords: *Time synchronization, data availability, data authentication, pair wise secure transaction, end-to-end delay.*

1. Introduction

Time synchronization plays a major role in Mobile Ad-hoc NETworks (MANETs). Because the movement of the nodes in MANET is not fixed, time synchronization helps the accurate, as well as secure transmission and other collaborative processes. For example, if an attack occurs and the time synchronization algorithm is affected, then the nodes will have faulty estimates of other nodes location, leading to chaos. While transmitting under these adverse conditions, packets might be lost or might be sent to wrong locations. Because of this, packet retransmissions occur based on the acknowledgements received. Collaborative data transmissions get affected. The

problems in time synchronization have been extensively studied in MANETs. Many algorithms have been introduced. But all these existing methods do not take into consideration the security issues. In our paper, we introduce security mechanisms into our basic method of sender-receiver synchronization for secure pair-wise time synchronization in MANETs. The overhead caused is minimum. This method can counter the attacks caused by external attackers. We extend our scheme for secure pair-wise synchronization over multiple hops also. Based on the requirements of the applications being used, for multiple hops, two schemes are introduced: opportunistic and direct. In Section 5 of the paper, we also discuss a mobility model and how it helps in assessing the data availability and storage of data. In MANETs the mobile nodes need to communicate with other nodes to share and access data. Due to disconnections, data from different MANETs cannot be accessed. Thus, data availability is deteriorated. Data replication and Data diffusion are two methods which can be used to solve the problem of data unavailability. In Section 6 we calculate the probability of providing data authentication to the data available.

2. Related works

Clock synchronization is a vital part for MANETs. Because in a MANET the nodes are mobile and their position is not fixed, time synchronization is difficult. Hence, a central coordinator is used to help in time synchronization. Previously, asynchronous clocks were mostly utilized. Due to this reason there is much power loss and there is an increase in the waiting time as the information of all nodes is not properly available to forward the packets. Herein, a protocol named MTSP is introduced for multi-hop MANETs. This protocol is for IEEE 802.11 mode. The MTSP consists of two phases: Beacon Window (BW) phase and SYNchronization (SYN) phase. Many protocols for clock synchronization in a MANET have been recently proposed. A time synchronism algorithm is proposed in [8] to deal with the partitioning problem in sparse ad hoc networks. RBS is presented in [9]. A reference broadcast does not contain an explicit timestamp; instead, its arrival time is used by the receivers as a point of reference for comparing their clocks. RBS uses nontrivial statistical methods, such as regression to estimate the clock phase offset and clock frequency offset of any two stations' protocol, called ASP, is proposed in [10] for time synchronization in 802.11-based multi-hop ad hoc networks. The basic idea of the protocol is to adjust clocks' frequencies. But it does not address the scalability problem fully. The maximum clock offset is still over 200 μs. The reason is that ASP trusts the face value of the timestamp of the beacon from the sending station. The result is that the slower station may over-adjust its clock frequency and become faster than the original fastest station. This process may keep repeating, the frequencies of the clocks getting faster and faster and eventually out of bound.

Various mobility models have also come into light in the recent times. The Random Waypoint Model was first proposed by J o h n s o n and M a l t z [5]. Soon it became a "benchmark" mobility model to evaluate the MANET routing protocols, because of its simplicity and wide availability. To generate the node trace of the Random Waypoint model, the setdest tool from the CMU Monarch group

may be used. This tool is included in the widely used network simulator ns-2. Another mobility model considering the temporal dependency of velocity over various time slots is the Smooth Random Mobility Model. In [6] it is also found that the memory-less nature of a Random Waypoint model may result in unrealistic movement behaviours. Instead of the sharp turn and sudden acceleration or deceleration, Bettstetter also proposes to change the speed and direction of the node movement incrementally and smoothly.

3. Problem formulation

In our network, we have MANET nodes. We assume that the nodes are neighbours or at a multiple hop distance. Here we accept that the links are bidirectional between the neighbouring nodes. We assume that the neighbouring nodes share pair-wise secret keys. We also assume that every node has a local clock.

Table 1

X, P	Sender nodes
Y, Q	Receiver nodes
R, S	Intermediate nodes
ack	Acknowledgement
sync	Synchronization
β	Offset (difference between two local clocks at a given time T)
D	End-to-end delay
D_{avg}	Average end-to-end delay
D_{actual}	Actual end-to-end delay
D_{M}	Maximum expected and allowed end-to-end delay
D^*	Maximum end-to-end delay
C_X	Local clock of node X
K_{XY}	Secret pair-wise key between nodes X and Y
t_{i1}, …, t_{i12}	Timestamps for sending or receiving packets between different nodes
Δ	Packet-delay
σ	Standard deviation

3.1. Sender-Receiver Synchronization

For two nodes to be synchronized, there are two main approaches: sender-receiver or receiver-receiver. We use the following algorithm for sender-receiver synchronization. In this paper we use the expression given below:

Node-id (Send time) → (Receive time) Node-id: contents of the packets.

Pair-wise Sender-receiver Synchronization:

1) $X(t_{i1}) \rightarrow (t_{i2})$ Y: X, Y, sync,
2) $Y(t_{i3}) \rightarrow (t_{i4})$ X: Y, X, t_{i2}, t_{i3}, ack,
3) X evaluates the offset.

Here t_{i1} and t_{i4} are the times calculated by the clock of X, C_X; t_{i2}, and t_{i3} are the times measured by C_Y. The offset represents the discrepancy between the local clocks at a given time T. The sender node X sends a synchronization packet to Y at time t_{i1}. At t_{i2} node Y receives the packet. Here t_{i2} is equivalent to $t_{i1}+\beta+D$, and β is the offset, D is the end-to-end delay. Node Y sends an acknowledgement back at t_{i3}.

This acknowledgement contains t_{i2} and t_{i3} values; t_{i4} is equivalent to $t_{i3}+\beta+D$. Thus, X calculates the end to end delay and offset as

(1) $\beta = \{(t_{i2} - t_{i1}) - (t_{i4} - t_{i3})\}/2$ and $D = \{(t_{i2} - t_{i1}) + (t_{i4} - t_{i3})\}/2$.

3.2. Malicious attacks

This subsection discusses the type of attacks an attacker can carry out externally to lead the pair-wise sender-receiver synchronization astray. If the attacker makes the nodes calculate the offset incorrectly:

A) t_{i2} and t_{i3} values can be modified. Thus the message to be sent can be changed or the identity of the receiving node can be changed.

B) The attacker can also influence the measurement of t_{i2} and t_{i3} instead of directly changing the values. The attacker can stop the first packet and then store it to replay to a further point which increases the transmission time. Thus, the offset at the sender's end can be modified. This is called packet-delay attack.

C) Similar packet-delay attack can be used to modify t_{i4} with the help of an acknowledgement packet.

If a packet-delay attack occurs, the equations change: $t_{i2}^* = t_{i1}+\beta+D$ and $t_{i4} = t_{i3} - \beta + D$, where $t_{i2}^* = t_{i2} + \Delta$. Here Δ represents the packet-delay that the attacker has introduced.

Thus, the clock offset and the end-to-end delay are changed to:

(2) $\beta = \{(t_{i2} - t_{i1}) - (t_{i4} - t_{i3}) + \Delta\}/2$ and $D = \{(t_{i2} - t_{i1}) + (t_{i4} - t_{i3}) + \Delta\}/2$.

In (2) β and D are increased by a factor of $\Delta/2$ than in (1).

4. Secure time synchronization

We introduce security aspects into sender-receiver synchronization to increase the resiliency of MANET towards malicious attacks from external attackers.

4.1. Secure Pair-wise Synchronization (SPS) Algorithm

The algorithm has three steps.

SPS Algorithm

Step 1. $X(t_{i1}) \rightarrow (t_{i2})$ Y: X, Y, sync.

Step 2. $Y(t_{i3}) \rightarrow (t_{i4})$ X: Y, X, t_{i2}, t_{i3}, ack, MAC$\{K_{XY}\}[Y, X, t_{i2}, t_{i3}$, ack].

Step 3. X calculates the end-to-end delay $D= \{(t_{i2} - t_{i1}) + (t_{i4} - t_{i3})\}/2$.

If $D \leq D^*$ then $\beta = \{(t_{i2} - t_{i1}) - (t_{i4} - t_{i3})\}/2$.

Else abort.

MAC (Message Authentication Codes) are used, and key K_{XY} that is shared between X and Y. Thus, with this algorithm, the integrity and the authenticity of the message are assured. The attackers cannot modify any values to cause a packet-delay attack. Node Y cannot be found by the attacker because the secret is held by X only. By comparing the actual end-to-end delay D, with the maximum expected end-to-end delay D^*, we can detect packet-delay attacks. Calculation of the offset is aborted if D is greater than D^*.

4.2. Performance evaluation of SPS

We can find out how great influence an attacker has on the synchronization based on the value of maximum end-to-end delay D^*. To understand better we calculate the value of D when there is no influence of external attackers. Thus, three main causes of the end-to-end delay are:

1. Waiting time at mac (medium access control) layer: This delay is not a fixed value.
2. The time that is taken to transmit the packet bit-by-bit from the sender node to the receiver node: This value can be determined and it depends on the transmission speed and size of the packet.
3. Propagation time from end to end: Its value ranges in nanoseconds.

4.2.1. Minimum synchronization precision

The nodes can be perfectly synchronized if the end-to-end delay is a fixed value. Then there would be a zero error. But because the delay keeps varying, a synchronization error occurs. The maximum error occurs when the end-to-end delay difference in both directions (from X to Y and from Y to X) is highest, i.e., in a given direction D is $D_{avg} - 3\sigma$ and in the opposite direction D is $D_{avg} + 3\sigma$. Here D_{avg} is the average delay and σ is the standard deviation. Thus, from the equations (1) and (2), we can conclude that the maximum synchronization error that can occur is 3σ.

4.2.2. Maximum attacker impact

This is defined as the maximum difference an attacker can cause between the clocks of two nodes without being caught. When the actual end-to-end delay is minimum, i.e. , $D_{avg} - 3\sigma$, then it is the worst case. Here the maximum packet-delay factor of 12σ ($\Delta = 12\sigma$). Node X will evaluate the end-to-end delay, D, as:

(2) $$D = D_{actual} + \Delta/2 = D_{avg} - 3\sigma + (12\sigma/2) = D_{avg} + 3\sigma = D^*.$$

From (2) we find that the offset will be reduced by $\Delta/2$. Hence, the maximum attacker impact is 6σ.

5. Multihop synchronization

Up to this moment we have considered synchronization between nodes which are directly connected (neighbours). Now we introduce two algorithms for increasing the security of sender-receiver synchronization which are multiple hops away from one another: opportunistic and direct. We presume that every pair of nodes has a path between them, formed based on the routing information and the topology of the network. For example, let us assume that we need to synchronize between nodes P and Q which have no direct communication. Let us presume that the shortest path betweenthe sender and the receiver is three hops away, going through R and S. Thus, the path is $P - R - S - Q$.

5.1. Secure Direct Multi-hop (SDM) Algorithm

The algorithm has five steps.

SDM Algorithm

Step 1. $P(t_{i1}) \rightarrow (t_{i2})\ R(t_{i3}) \rightarrow (t_{i4})\ S(t_{i5}) \rightarrow (t_{i6})\ Q$: P, Q, sync.

Step 2. Q: $h1 = \{Q, P, t_{i6}, t_{i7}, \text{ack}\}$

: $H1 = \text{MAC}\{K_{QS}\}[Q, S, h1]$

$Q(t_{i7}) \rightarrow (t_{i8})\ S$: Q, S, $h1$, $H1$.

Step 3. S: $h2 = \{Q, S, P, t_{i4}, t_{i9}, (t_{i6} - t_{i5}), (t_{i8} - t_{i7}), \text{ack}\}$

: $H2 = \text{MAC}\{K_{SR}\}[S, R, h2]$

$S(t_{i9}) \rightarrow (t_{i10})\ R$: S, R, $h2$, $H2$.

Step 4. R: $h3 = \{Q, S, R, P, t_{i2}, t_{i11}, (t_{i4} - t_{i3}), (t_{i10} - t_{i9}), (t_{i6} - t_{i5}), (t_{i8} - t_{i7}), \text{ack}\}$

: $H3 = \text{MAC}\{K_{RP}\}[R, P, h3]$

$R(t_{i11}) \rightarrow (t_{i12})\ P$: R, P, $h3$, $H3$.

Step 5. P: calculate

$D = \{[(t_{i2} - t_{i1}) + (t_{i4} - t_{i3}) + (t_{i6} - t_{i5})] + [(t_{i12} - t_{i11}) + (t_{i10} - t_{i9}) + (t_{i8} - t_{i7})]\}/2$

if $D < D_T$ * then

$\beta = \{[(t_{i2} - t_{i1}) + (t_{i4} - t_{i3}) + (t_{i6} - t_{i5})] - [(t_{i12} - t_{i11}) + (t_{i10} - t_{i9}) + (t_{i8} - t_{i7})]\}/2$.

Else abort.

The sender and the receiver need not share the secret key, i.e., P and Q need not share the pair-wise secret key. But P and R, R and S, S and Q need to share the pair-wise secret key. To understand the algorithm, first we study the transmission between neighbours P and R.

(3) $$P(t_{i1}) \rightarrow (t_{i2})\ R;\ R(t_{i11}) \rightarrow (t_{i12})\ P.$$

These four timestamps are interrelated as follows:

(4) $$t_{i2} = t_{i1} + \beta_{PR} + D_{PR};\ t_{i12} = t_{i11} - \beta_{PR} + D_{PR.}$$

Similarly for the pairs (R, S) and (R, Q) we obtain the following relationships between their timestamps:

(5) $$t_{i4} = t_{i3} + \beta_{RS} + D_{RS};\ t_{i10} = t_{i9} - \beta_{RS} + D_{RS},$$

(6) $$t_{i6} = t_{i5} + \beta_{RQ} + D_{RQ};\ t_{i8} = t_{i7} - \beta_{RQ} + D_{RQ}.$$

Combining mathematically (5) and (6), and including the terms β_{PQ} and D_{PQ}, the following equations are produced:

(7) $$(t_{i2} - t_{i1}) + (t_{i4} - t_{i3}) + (t_{i6} - t_{i5}) = \beta_{PQ} + D_{PQ},$$

(8) $$(t_{i12} - t_{i11}) + (t_{i10} - t_{i9}) + (t_{i8} - t_{i7}) = -\beta_{PQ} + D_{PQ}.$$

Here $\beta_{PQ} = \beta_{PR} + \beta_{RS} + \beta_{SQ}$ and $D_{PQ} = D_{PR} + D_{RS} + D_{SQ}$.

5.2. Performance evaluation of SDM

Secure Direct Multi-hop synchronization (SDM) is better than SOM in terms of accuracy. However, in SDM the intermediate nodes are by default assumed to be trustworthy. Thus, SDM is not resilient enough to secure itself from attacks by compromised nodes. In SDM, mac access delays do not affect the end-to-end delay. Hence, based on the number of hops, we can easily evaluate the end-to-end delay. The end-to-end delay is equivalent to the collective sum of D_{PR}, D_{RS} and D_{SQ}.

6. Conclusion

In this paper we have introduced a time synchronization technique for secure transmission in case of attacks in MANETs. In this technique we have used algorithms for pair-wise secure transactions for neighbouring nodes, as well as nodes separated by multiple hops. Data replication and Data diffusion are two methods which are used to solve the problem of data unavailability. Future research will mainly concentrate on integrating the better security concepts for data authentication and synchronization.

References

1. Fan, X., Y. Song. Improvement on Leach Protocol for Wireless Sensor Network. – In: SensorCom'2007, 2007, 260-264.
2. Heinzelman, W., A. Chandrakasan, H. Balakrishnan. An Application Specific Protocol Architecture for Wireless Microsensor Networks. – IEEE Trans. Wireless Commul1., Vol. **1**, October 2002, No 4, 660-670.
3. Kumari, Anu, Arvind Kumar, Akhil Sharma. Survey Paper on Energy Efficient Routing Protocol in MANET. – International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 3, March 2013, No 3, 29-33.
4. Zhu, Jinhua, Wang Xin. Model and Protocol for Energy-Efficient Routing over Mobile Ad-Hoc Networks. – IEEE Transactions on Mobile Computing, Vol. **10**, November 2011, No 11, 1546-1557. doi:10.1109/TMC.2010.259.
5. Kang, Seokhoon, Gwanggil Jeon, Young-Sup Lee. Improved Energy Aware Routing Protocol in Mobile Ad-Hoc Network. – In: 6th International Conference, ICHIT 2012, Daejeon, Korea, 23-25 August 2012, 106-113.
6. Chang, J-H, L. Tassiulas. Energy Conserving Routing in Wireless Ad-Hoc Networks. – In: Proceedings of the Conf. on Computer Communications (IEEE Infocom'2000), 2000, 22-31.
7. Li, Q., J. Aslam, D. Rus. Online Power-Aware Routing inWireless Ad-Hoc Networks. – In: Proceedings of Int. Conf. on Mobile Computing and Networking (MobiCom'2001), 2001.
8. Doshi, S, T. X. Brown. Minimum Energy Routing Schemes for a Wireless Ad-Hoc Network. – In: Proceedings of the Conference on Computer Communications (IEEE Infocom'2002), 2002.
9. Chen, Y. R., L. Yu, Q. F. Dong, Z. Hong. Power Control in Wireless Sensor Network Based on Nearest Neighbor Algorithm. – J. Zhejiang Univ. Eng Sci., Vol. **44**, 2010, 1321-1326.
10. Heinzelman, W., R. A. Chandrakasan, H. Balakrishnan. Energy-Efficient Communication Protocol for Wireless Miceosensor Networks. – In: Proceedings of the 33rd Hawai International Conference on System Science, 4-7 January 2000, Washington, DC, USA. PP8020-8025.