IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 4, April 2018

RET-CAN: Robust and Efficient Transmission in CAN

P. Mathiazhagan[1], Dr. A. Marimuthu[2]

Research Scholar, Government Arts College, Coimbatore, India[1]

Assosiate Professor, Government Arts College, Coimbatore[2]

Abstract: Controller Area Network (CAN)-based networked system. In these systems, the network is the central information hub, on which time critical data are transmitted. CAN provides an inexpensive, durable network that helps multiple CAN devices communicate with one another. An advantage to this is that electronic control units (ECUs) can have a single CAN interface rather than analog and digital inputs to every device in the system. This decreases overall cost and weight in automobiles. An energy efficient clustering algorithm with optimum parameters is designed for reducing the energy consumption and prolonging the system lifetime. Here we approach an Energy Efficient Clustering Mechanism (EECM) with Firefly algorithm. And also present a Cluster key Management for secure transmission in WSN. A Firefly Algorithm (FA) is a recent nature inspired optimization algorithm that simulates the flash pattern and characteristics of fireflies. Clustering is a popular data analysis technique to identify homogeneous groups of objects based on the values of their attributes. And we propose an efficient Cluster Key Management (CKM) scheme for secure communication in dynamic WSNs characterized by node mobility. The CKM supports efficient key updates when a node leaves or joins a cluster and ensures forward and backward key secrecy. We plan to improve the performance of the proposed security protocol with an implementation of the encryption to optimize our security technology. And demonstrated it through simulation experiments.

Keywords: CAN, ECU, CKM, EECM.

I. INTRODUCTION

CAN networks, called Controller Area Networks, can be used in the framework of real-time distributed industrial applications. Such applications cover manufacturers, the distributed and parallel processing systems in industrial and networking fields. The electronic control unit (ECU) is the most essential component that controls one or more of the electrical systems and subsystems in a vehicle. State-of-the art vehicular on-board architectures can consist of more than 70 ECUs that are interconnected via heterogeneous communication networks such as the controller area network (CAN). CAN networks guarantee sufficiently short time latency and it has been shown that these systems exceed in performance to the token-based ones. Access to the medium in wired CAN is shared based. It respects the CSMA/CA scheme which is “Arbitration on Message Priority” and “bit-wise Contention” technique. This technique, along with the mechanism of detecting and correcting errors, gives high performance to the protocol CAN to be adopted for real-time applications where multiple access are applied. CAN protocol is a message-based or data-centric protocol, in which, messages are not transmitted from one node to another based on addresses. Instead, all nodes in the network receive the transmitted messages in the bus and decide whether the message received is to be discarded or processed. Depending on the system, a message may destine to either one or many nodes. This has several important consequences such as system flexibility, message routing and filtering, multicast, together with data consistency.To check the functions of the ECUs during a diagnostic process, the tools broadcast CAN data frames without encryption and authentication to force control of the ECUs. This means that an adversary can also use an automotive diagnostic tool to easily get CAN data frames that can control an ECU. We note that a security technique used for the general IT environment cannot be immediately applied to CAN, as it has unique features such as a limited data payload. Therefore, it is necessary to design an efficient security technique.The previous work demonstration of a practical long-range wireless attack experiment using a malicious smartphone app in a connected car environment. Design of a security protocol that can be implemented on an ECU, accommodating the limited resources available and the current CAN data frame format. Analysis of the security and performance of the proposed security protocol using Secure-ECU and CANoe.This decreases overall cost and weight in automobiles. Hence, the reliability of the network not only has a direct impact on the system performance but also affects the safety of the system operations.In this proposed work we used an Energy Efficient Clustering Mechanism (EECM) with Firefly algorithm. And also present a Cluster key Management for secure transmission in WSN. A Firefly Algorithm (FA) is a recent nature inspired optimization algorithm that simulates the flash pattern and characteristics of fireflies. Clustering is a popular data analysis technique to identify homogeneous groups of objects based on the values of their attributes. We use the firefly algorithm to find initial optimal cluster

DOI 10.17148/IJARCCE.2018.748

centroid and then optimized centroid to refined them and improve clustering accuracy. And we propose an efficient Cluster Key Management (CKM) scheme for secure communication in dynamic WSNs characterized by node mobility. The CKM supports efficient key updates when a node leaves or joins a cluster and ensures forward and backward key secrecy.

II. CAN PROTOCOL

The controller area network (CAN) was created by Robert Bosch in mid 1980s as a new vehicle bus communication between the control units in automobile industries. In the past the vehicle bus communication used point to point communication wiring systems which caused complexity, bulkiness and heavy and expensive with increasing electronicsand controller deployed vehicles. According to Fig.1 theabundance of wiring required makes the whole circuit complicated. CAN solves this complexity by using twisted pair cables that is shared throughout the control units which can be seen in Fig.2. Not only does it reduce the wiring complexity but it also made it possible to interconnect several devices using only single pair of wires and allowing them to have simultaneous data exchange.

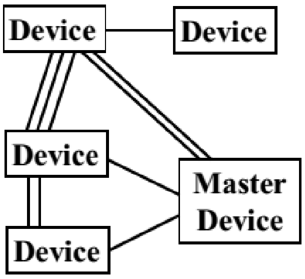


Fig.1. Traditional Wiring Method

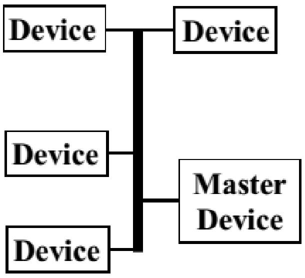


Fig.2. CAN Wire Method

III. RELATED WORK

The invehicle CAN, which was regarded as a closed network inthe past, is now being connected to external networks andprovides useful services such as Remote Diagnostics, Firmware Updates Over the Air, and Real-timeProduct Carbon Footprints Data Analysis. On the otherhand, such connectivity to external networks introduces a newtype of security threat to the vehicle.Koscher et al. suggested specific wireless attack techniquesand experimented with short- and long-range wireless attacks. A short-range wireless attack is possible when a Bluetooth device installed on the vehicle is paired with the driver'ssmartphone on which a malicious app has been installed. Along-range wireless attack is possible owing to the vulnerability of the authentication function in the aqLink protocol.However, to conduct the wireless attacks in, complex andadvanced technologies such as reverse engineering are requiredto analyze automotive electronics. In addition, the long-rangewireless attack is possible only for a vehicle using the aqLinkprotocol. The previous studies on vehicular security point outvulnerabilities of the in-vehicle CAN as the primary cause of acyber attack. In particular, mentions the lack of data frame authentication and encryption as the most severe vulnerabilities of CAN.To provide an in-vehicle CAN communication environment secure against a replay attack, and proposed data authentication techniques that considered the limited data payload of a CAN data frame. Groza and Marvay suggested a CAN data authentication protocol using a TESLA-like protocol in. In TESLA, a sender attaches to each data a MAC computed with a key k known only to the sender. A short time later, the sender sends k to the receiver, who can then authenticate the data. We note that key disclosure delay in the TESLA-like protocol should be minimized to ensure real-time processing in CAN. However, the shorter the delay is, the larger the bus load is. Our simulation in the next section shows that the TESLA-like protocol finds it difficult to provide real-time processing in CAN. Groza et al. also proposed a single master case to minimize key disclosure delay. In a single master case, the sender generates a MAC

DOI 10.17148/IJARCCE.2018.748

with a long-term secret key shared with the communication master and transmits a data and the corresponding MAC to the master. The master then transmits the data andMAC to the receivers. However, since the secret key sharedbetween a sender and the communication master cannot be Changed for each session, a replay attack after eavesdropping on the transmitted CAN data frame and MAC is possible. Lin et al. proposed a MAC generation technique using an ID table, message counter, and pair-wise symmetric key (PWSK). Receivers' IDs are registered on a sender's ID table. It is assumed that a sender shares a PWSK with the receivers in the ID table. Their MAC generation technique is similar to ours in that it uses a synchronized message counter among ECUs. However, the protocol of Lin et al. uses a PWSK, whereas ours uses a group session key. Using a PWSK implies that a sender must generate as many MACs as receivers in the communication group and transmit them separately to each receiver. This will increase the bus load rapidly and is hence impractical. In addition, their security technique does not consider data confidentiality and connectivity with external devices.

VI. PROPOSED METHODOLOGY

In this framework we show that a long-range wireless attack is physically possible using a real vehicle and malicious smartphone application in a connected car environment. We use a security protocol for CAN as a countermeasure designed in accordance with current CAN specifications. We demonstrate a practical wireless attack using a real vehicle in a connected car environment, in which a driver's smartphone is connected to the in-vehicle CAN.

Energy Efficient Clustering Mechanism (EECM) with Firefly algorithm. And also present a Cluster key Management for secure transmission in WSN. A Firefly Algorithm (FA) is a recent nature inspired optimization algorithm that simulates the flash pattern and characteristics of fireflies. Clustering is a popular data analysis technique to identify homogeneous groups of objects based on the values of their attributes. We use the firefly algorithm to find initial optimal cluster centroid and then optimized centroid to refined them and improve clustering accuracy. And we propose an efficient Cluster key management (CKM) scheme for secure communication in dynamic WSNs characterized by node mobility. The CKM supports efficient key updates when a node leaves or joins a cluster and ensures forward and backward key secrecy.

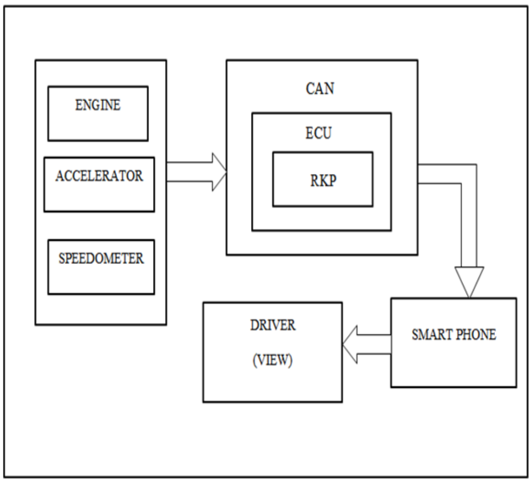


Fig.3 Framework of Security Based Protocol Design in CAN.

A. Firefly Algorithm (FA)
The FA process:

- In this work, we use firefly algorithm, A Firefly Algorithm (FA) is a recent nature inspired optimization algorithm that simulates the flash pattern and characteristics of fireflies.
- Clustering is a popular data analysis technique to identify homogeneous groups of objects based on the values of their attributes.
- Firefly algorithm is a swarm based algorithm that use for solving optimization problems. This paper presents a new approach to using firefly algorithm to cluster data. It is shown how firefly algorithm can be used to find the centroid of the user specified number of clusters.
- We use the firefly algorithm to find initial optimal cluster centroid and then optimized centroid to refined them and improve clustering accuracy.
- Easy and efficient implementation, Easy to understand and Parallel implementation. This method helps to prolong the network lifetime.

DOI 10.17148/IJARCCE.2018.748

i. Initialization

The first step in the algorithm is the initialization of the population of N fireflies where each firefly represents a candidate solution. Population size (N) represents the number of solutions or the size of the search space. An objective function is associated with the brightness of the firefly and is directly proportional to the brightness. The aim is to maximize the objective function value.

ii. Firefly evaluation

Firefly algorithm is based upon idealizing the flashing characteristic of fireflies. The idealized three rules are:-
All fireflies are considered as unisex and irrespective of the sex one firefly is attracted to other fireflies. The Attractiveness is proportional to their brightness, which means for any two flashing fireflies, the movement of firefly is from less bright towards the brighter one and if no one is brighter than other it will move randomly. Furthermore they both decrease as their distance increases. The landscape of the objective function directly affects the brightness of the firefly.

iii. Distance calculation

The distance between any two fireflies i and j at xi and xj respectively, the Cartesian distance is determined by equation where xi, k is the k th component of the spatial coordinate xi of the i th firefly and d is the number of dimensions.

$$d_{i,j} = \text{Distance}(\mathbf{x}^i, \mathbf{x}^j) = \sqrt{\sum_{k=1}^{n} (x_k^i - x_k^j)^2}$$

iv. Attractiveness

In the Firefly algorithm, there are two important issues: the variation of the light intensity and the formulation of the attractiveness. We know, the light intensity varies according to the inverse square law. Suppose it is absolute darkness. Light intensity of each firefly is proportional to quality of solution. Each firefly needs to move towards the brighter fireflies. Light intensity reduction abides the law:

(I_0, d)=I_0/d^2

I_0 is the light intensity at zero distanced
d is the observer’s distance from source

If we take absorbtion coefficient "ɣ" into account:
Attractiveness (I_0, d, "ɣ") = I_0 e^(−"ɣ" d^2)
Ir = Isr 2 (1)

Where I(r) is the light intensity at a distance r and Isis the intensity at the source.
When the medium is given the light intensity can be determined as follows:
Ir = I0e −γr(2)
To avoid the singularity at r=0 in (1), the equations can be approximated in the following Gaussian form:
Ir = I0e −γr 2 (3)

As we know, that a firefly’s attractiveness is proportional to the light intensity seen by adjacent fireflies and thus the attractiveness β of a firefly is determined by equation (4) where β0 is the attractiveness
atr=0. β = β0e–γrm(m≥1) (4)

v. Movement

The movement of a firefly i is attracted to another more attractive (brighter) firefly j is determined by

xi = xi + β0e −γrij 2 xj − xi + αE

Movement consist two elements

- Approach to better solutions
- Move randomly

Pseudo code for Firefly Algorithm

```
1. Objective function of f(x),
2. Generate initial population of fireflies;
3. Formulate light intensity I;
4. Define absorption coefficient γ;
5. While (t Ii), move firefly i towards j;
6. For i = 1 to n (all n fireflies);
7. For j=1 to n (all n fireflies)
```

8. If (Ij> Ii), move firefly i towards j;
9. end if
10. Evaluate new solutions and update light intensity;
11. End for j;
12. End for i;
13. Rank the fireflies and find the current best;
14. End while;
15. Post process results and visualization;
16. End procedure

B. Key Management

We consider a dynamic wireless sensor network (See Fig. 2). The network consists of a number of stationary or mobile sensor nodes and a BS that manages the network and collects data from the sensors. Sensor nodes can be of two types: (i) nodes withhigh processing capabilities, referred to as H-sensors, and (ii) nodes with low processing capabilities, referred to as L-sensors.Nodes may join and leave the network, and thus the network size may dynamically change. The H-sensors act ascluster headswhileL-sensors act as cluster members. They are connected to the BS directly or by a multi-hop path through otherH-sensors.H-sensors andL-sensors can be stationary or mobile. After the network deployment, eachH-sensor forms a cluster by discovering the neighboring L-sensors through beaconmessage exchanges. TheL-sensors can join a cluster, move to other clusters and also re-join the previous clusters. To maintain the updated list of neighbors and connectivity, the nodes in a cluster periodically exchange very lightweight beaconmessages. TheH-sensors report any changes in theirclusters to the BS, for example, when a L-sensor leaves or joins the cluster. The BS creates a list of legitimate nodes, M, and updates the status of the nodes when an anomaly node or node failure is detected. The BS assigns each node a unique identifier. A L-sensor nLi is uniquely identified bynode IDLi whereas aH-sensornHj is assigned a node ID Hj. A Key Generation Center (KGC), hosted at the BS, generatespublic system parameters used for key management by the BS and issues certificateless public/private key pairs for each node in the network. In our key management system, a unique individual key, shared only between the node and the BS is assigned to each node. The certificateless public/private key of a node is used to establishpairwise keys between any two nodes. Acluster keyis shared among the nodes in a cluster.

In this section, we propose a Cluster Key Management scheme (CKM) that supports the establishment of four types of keys, namely: a certificateless public/private key pair, an individual key, a pairwise key, and a cluster key. This scheme also utilizes the main algorithms of the CKM scheme in deriving certificateless public/private keys and pairwise keys.
Types of Keys

- Certificateless Public/Private Key: Before a node is deployed, the KGC at the BS generates a unique certificateless private/public key pair and installs the keys in the node. This key pair is used to generate a mutually authenticated pairwise key.
- Individual Node Key: Each node shares a uniqueindividual key with BS. For example, a L-sensor can usethe individual key to encrypt an alert message sent tothe BS, or if it fails to communicate with the H-sensor.AnH-sensor can use its individual key to encrypt themessage corresponding to changes in the cluster. The BS can also use this key to encrypt any sensitive data, such as compromised node information or commands. Before a node is deployed, the BS assigns the node the individual key.
- Pairwise Key:Each node shares a different pairwise keywith each of its neighboring nodes for secure communications and authentication of these nodes. For example, in order to join a cluster, aL-sensor should share a pairwise key with theH-sensor. Then, the H-sensor can securely encrypt and distribute its cluster key to the L-sensor by using the pairwise key. In an aggregation supportive WSN, theL-sensor can use its pairwise key to securely transmit the sensed data to the H-sensor. Each node can dynamically establish the pairwise key between itself and another node using their respective certificateless public/private key pairs.
- Cluster Key:All nodes in a cluster share a key, named ascluster key. The cluster key is mainly used for securing broadcast messages in a cluster, e.g., sensitive commandsor the change of member status in a cluster. Only the cluster head can update the cluster key when aL-sensor leaves or joins the cluster.

Key Identifier Set Generation

Create a one-dimensional array of length and initialize the array with “0”s. After the following steps, output the array which is fulfilled with key identifiers for a node. For each
(1) The LFSR shift to producing bit stream;
(2) Once every bits have been generated, compute value ;
(3)if or , go to (1);

(4)else, ;
(5) Sort the sequence using standard insertion sort method end for.

Security

Resilience in WSNs refers to the resistance of key distribution schemes against node capture. When sensor nodes are deployed in hostile areas (e.g., battle surveillance), an adversary can mount a physical attack on a sensor node and recover secret information from its memory. So we are interested in the question: for any two nodes and which have not been captured by the attacker, what is the probability that the attacker can eavesdrop their communications using the subset of the key pools that was recovered from the nodes captured. That is, the shared key of two physically connected neighbors (also called a link) is in the compromised key set or a path key is compromised in path key establishment.

Security Algorithm

```
1. Node (N) sends a RREQ to Neighbor Node (NN).
2.Neighbor Node (NN) receives the RREQ from Node (N) and RRES to Node (NN).
//Trust Calculation//
3.Neighbor information is gathered and sensed,
i.      Energy.
ii.     Packet Count.
iii.    Queue Size.
4.      It generates a report and validate the report rules.
5.      The trust value is calculated using,
Tc= ts + (p/2) / (t+P)
ts, t >= 0, p>0
Where, Tc is the trust calculation, L represents direct trust, ts illustrates the time success, t is the time transactions, and p is the positive real number.
6.      The current trust value (C_TV) is retrieved.
if(C_TV > 0.7)
{
if(selfish node is detected)
Add selfish node to block list (BL);
else
Transfer the data to destination node;
}
7. Finally, the performance is evaluated.
```

Complexity

In this section, the storage, computation, and communication complexity of the new scheme are presented.

(1) Storage Complexity. Same as the basic scheme, each node in the new scheme stores a key ring and a neighbor list. There are key-identifier/key pairs in the key ring. Let represent the length of a key, so the key ring takes bits of storage space. On the other hand, the probability that two neighbor nodes are logically connected is also , and the average number of neighbors for a node is . Thus, there are about records on the neighbor list. Each record includes a node identifier and a key identifier. Therefore, the neighbor list uses bits memory space. Consequently, the total storage complexity of the new scheme is .

(2) Communication Complexity. Let be the length of a ciphertext. If and share a key, the main message transmitted on the channel is the ciphertext and the key identifier of the encryption key. So, the communication complexity is , where represents the number of hops in a path.

Simulation

In this section, we use computer to simulate the basic scheme and our scheme. In the basic scheme, every hop of the path is not only physically but also logically connected. We try to estimate the amount of computation on deciding whether two adjacent nodes are logically connected (neighbor list query) in routing process. On the other hand, we prove that the path length between two nodes will increase in the basic scheme.

(1) Computation Amount. As previously metioned, the basic scheme will spend lots of extra computation on neighbor list query in the routing process.
(2) Path Length. In fact, the path length of the basic scheme may be longer than that of the new scheme since the routing protocol is affected by the basic scheme. Consequently, the communication complexity of the basic scheme will be more than the communication amount.

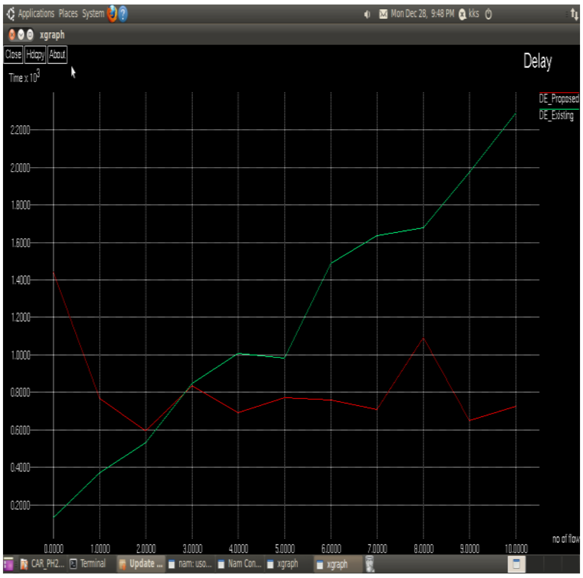
DOI 10.17148/IJARCCE.2018.748

(3) Execution Time of the Path Key Establishment. We simulate the path key establishment process on the computer, in which the encryption algorithm is DES.

V. PERFORMANCE ANALYSIS

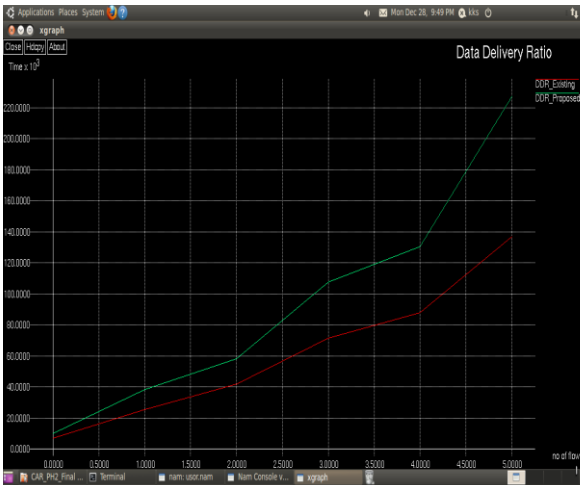
In this section, we evaluate our optimization algorithms using a popular WSN simulation platform. Here we analysis about the Data delivery ratio, trust model and transmission delay ratio.

End-to-End Delay:



The above comparison is described the Delay in our proposed and existing models. In our existing having more amount of delay to compared proposed method. In this proposed we achieved less amount of delay.Proposed method we improved our energy efficiency, energy consumption in order to we decreased our end-to-end transmission and packet forwarding delay. Above all analysis gives better performance compared to existing methods. In our proposed model we achieve more energy efficiency, less energy consumption, decreased delay and more security.

Data Delivery ratio:

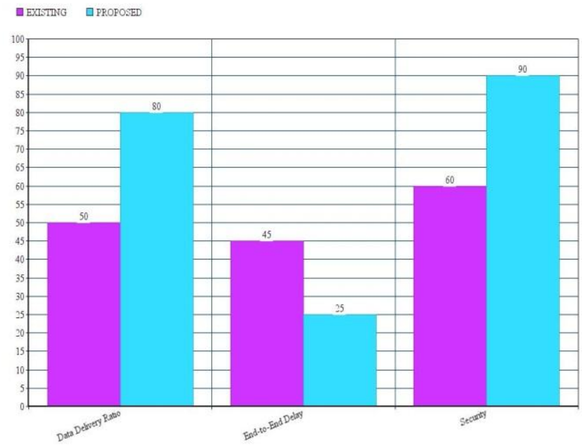


The above picture shows the data delivery ratio of our proposed model.
This comparison between existing transmission and trust model data delivery ratio and proposed delivery model ratio.
Here in proposed method having high efficiency of delivery ratio compare to our existing method delivery ratio.
We got high data ratio that means min level of energy used and more energy saved in our proposed method. Compared to the proposed method in existing energy level usage is high. So in our proposed we decreased energy usage and increased energy efficiency ratio.

DOI 10.17148/IJARCCE.2018.748

Comparison Chart:

Here we compare our performance analysis results in comparison chart. Compared existing and proposed results, the parameters are Data delivery ratio, End-to-End delay and security ratio. In below we have chart for these comparison.



VI. CONCLUSION AND FUTURE WORK

In this work,we propose a security protocol for CAN as a countermeasure designed in accordance with current CAN specifications. It is an actual attack model using a malicious smartphone app in the connected car environment. In This method we used Energy Efficient Clustering Mechanism (EECM) with Firefly algorithm. And also present a Cluster key Management for secure transmission in WSN. A Firefly Algorithm (FA) is a recent nature inspired optimization algorithm that simulates the flash pattern and characteristics of fireflies. And we propose an efficient Cluster key management (CKM) scheme for secure communication in dynamic WSNs characterized by node mobility.

The CKM supports efficient key updates when a node leaves or joins a cluster and ensures forward and backward key secrecy.Here the high data delivery ratio, less delay takes place a and also our network lifetime increases. The Cryptographic key management method can provide better security, scalability than any other previous methods. In the future, we will continue the work and apply our CAN optimization to other networking systems which face the similar trust conflict like WSNs, such as the vehicle ad hoc networks and the anonymity networks.

REFERENCES

[1] A. Saad and U. Weinmann, “Automotive software engineering and concepts,” GI. Jahrestagung., vol. 34, pp. 318–319, 2003.

[2] E. Nickel, “IBM automotive software foundry,” in Proc. Conf. Comput. Sci. Autom. Ind., Frankfurt, Germany, 2003.

[3] M. Wolf, A. Weimerskirch, and T. Wollinger, “State of the art: Embedding security in vehicles,” EURASIP J. Embedded Syst., vol. 2007, no. 5, p. 1,2007.

[4] R. Charette, This Car Runs on Code. [Online]. Available: http://www. spectrum.ieee.org/feb09/7649

[5] T. Nolte, H. Hansson, and L. L. Bello, “Automotive communications-past, current and future,” in Proc. IEEE Int. Conf. Emerging Technol. Factory Autom., 2005, vol. 1, pp. 992–999.

[6] K. H. Johansson, M. Torngren, and L. Nielsen, “Vehicle applications of controller area network,” in Handbook of Networked and Embedded Control Systems. New York, NY, USA: Springer-Verlag, 2005, pp. 741–765.

[7] T. Hoppe and J. Dittman, “Sniffing/replay attacks on CAN buses: A simulated attack on the electric window lift classified using an adapted CERT taxonomy,” in Proc. Conf. Embedded Syst. Security, 2007, pp. 1–6.

[8] T. Hoppe, S. Kiltz, and J. Dittmann, “Security threats to automotive CAN networks—Practical examples and selected short-term countermeasures,” Rel. Eng. Syst. Safety, vol. 96, no. 1, pp. 11–25, Jan. 2011.

[9] [9] K. Koscher et al., “Experimental security analysis of a modern automobile,” in Proc. IEEE Security Privacy. Symp., Oakland, CA, USA, 2010, pp. 447–462.

[10] The EVITA project, 2008, Webpage. [Online]. Available: http:// evita-project.org

[11] H. Schweppe, Y. Roudier, B. Weyl, L. Apvrille, and D. Scheuermann, “Car2X communication: Securing the last meter—A cost-effective approach for ensuring trust in Car2X applications using in-vehicle symmetric cryptography,” in Proc. Conf. Veh. Technol., San Francisco, CA, USA, 2011, pp. 1–5.

[12] H. Schweppe et al., “Securing Car2X applications with effective hardware software codesign for vehicular on-board networks,” in Proc. Conf. Autom. Security, Berlin, Germany, 2011.

[13] D. K. Nilsson, U. E. Larson, and E. Jonsson, “Efficient in-vehicle delayed data authentication based on compound message authentication codes,” in Proc. Conf. IEEE 68th Int. Conf. Veh. Technol., Calgary, BC, Canada, 2008, pp. 1–5.

[14] B. Groza and S. Murvay, “Efficient protocols for secure broadcastin controller area networks,” IEEE Trans. Ind. Informa., vol. 9, no. 4, pp. 2034–2042, Nov. 2013.

[15] C. W. Lin and A. SangiovanniVincentelli, “Cyber-security for the Controller Area Network (CAN) communication protocol,” in Proc. Conf. IASE Int. Conf. Cyber Security, 2012, pp. 344–350.

[16] P. Kleberger, T. Olovsson, and E. Jonsson, “Security aspects of the invehicle network in the connected car,” in Proc. IEEE Intell. Veh.,Symp., 2011, pp. 528–533.

[17] BOSCH CAN, 2004, Webpage. [Online]. Available: www.can.bosch.com[18] D. K. Nilsson, U. E. Larson, and E. Jonsson, “Creating a secure infrastructure for wireless diagnostics and software updates in vehicles,” in Proc. Conf. Comput. Safety, Rel., Security, Tyne, UK., Newcastle upon Tyne, U.K., 2008, pp. 207–220.

[18] S. Checkoway et al., “Comprehensive experimental analyses of automotive attack surfaces,” in Proc. 19th Conf. USENIX Sec., Washington, DC, 2011, p. 6.

[19] IEEE Standard for Local and Metropolitan Area Networks Part 16 Air Interface for Fixed and Mobile Broadband Wireless Access Systems, IEEE Std 802.16, 2009, IEEE Standard.

[20] M. Bellare and P. Rogaway, “Entity authentication and key distribution,” in Proc. Conf. CRYPTO, 1993, pp. 232–249.

[21] J. M. Alfred, P. C. van O, and A. V. Scott, Handbook of Applied Cryptography, Chapter-9-Hash Function. Boca Raton, FL, USA: CRC Press, 1997, pp. 359–368, no. 4.

[22] S. You, M. Krage, and L. Jalics, “Overview of remote diagnosis and maintenance for automotive systems,” in Proc. SAE World Congr., Detroit, MI, USA, 2005, pp. 1–8.

[23] M. Shavit, A. Gryc, and R. Miucic, “Firmware Update Over The Air (FOTA) for automotive industry,” in Proc. Conf. Asia Pacific Autom. Eng., Hollywood, CA, USA, 2007.

[24] D. K. Nilsson and U. E. Larson, “Secure firmware updates over the air in intelligent vehicles,” in Proc. IEEE Int. Conf. Commun. Workshop, Beijing, China, 2008, pp. 380–384.

[25] H. Hilpert, L. Thoroe, and M. Schumann, “Real-time data collection for product carbon footprints in transportation processes based on OBD2 andsmartphones,” in Proc. Conf. Syst. Sci., 2011, pp. 1–10.

[26] J. Daemen and V. Rijmen, The Design of Rijndael. AES-the Advanced Encryption Standard. Berlin, Germany: Springer-Verlag, 2002.

[27] K. Yasuda, “Multilane HMAC: Security beyond the birthday limit,” in Proc. Conf. INDOCRYPT, 2007, pp. 18–32.

[28] A. Hodjat and I. Verbauwhede, “Minimum area cost for a 30 to 70 Gbits/s AES processor,” in Proc. IEEE. Comput. Soc. Annu. Symp VLSI, 2004, pp. 83–88.

[29] S. Mangard, M. Aigner, and S. Dominikus, “A highly regular and scalable AES hardware architecture,” IEEE Trans. Comput., vol. 52, no. 4, pp. 483–491, Apr. 2003.

[30] Vector, Webpage. [Online]. Available: www.vector-informatik.com

[31] Texas Instruments, Webpage. [Online]. Available: http://www.ti.com/TMS570.

DOI 10.17148/IJARCCE.2018.748