ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

Detection and Filter of False Data using RC6 Encryption Mechanism

Aruna N S[1], Anu C S[2]

Assistant Professor, Department of CSE, Global Academy of Technology, Bengaluru, India[1]

Assistant Professor, Department of CSE, Bapuji Institute of Engineering & Technology, Davanagere, India[2]

ABSTRACT:Today most of the applications such as military operations, medical systems, aerospace systems, robotic systems, process control, factory automation, building and environmental control and smart spaces need the integration of the sensor-based systems, hence there is a need to provide security for these sensor based applications for the healthy operations. It is very important to provide authentic and accurate data to surrounding sensor nodes. Here we are introducing a Optimized Energy Management Using Advanced Cryptographic Algorithm for Wireless Sensor Network(OEMAC), which efficiently detect and filter false data injected into the network by malicious outsiders. OEMAC uses RC6 algorithm. The key to the RC6 encryption mechanism dynamically changes as a function of the residual energy of the node. Thus, a one-time dynamic key is employed for one packet only and different keys are used for the successive packets of the stream. Thus it consumes less energy and provides more security.

I. INTRODUCTION

In the recent past, wireless sensor networks have found their way into a wide variety of applications and systems with vastly varying requirements and characteristics. Smart environments represent the next evolutionary development stepin building, utilities, industrial, home, shipboard, and transportation systems automation. The wireless sensor networks are playing a key role in variety of application scenarios. The typical application scenarios include environmental, military, and commercial enterprises[3]. In another aspect, the underwater sensor nodes are very useful in oceanographic data collection, pollution monitoring, assisted navigation, military surveillance, and mine reconnaissance operations. The improvements in technology will give more sensor applications in our daily lives [5].
Time critical responses such as troop movement, evacuation, and first response deployment In this paper, we focus on the keying mechanisms for Wireless Sensor Networks. Securing the diverse set of sensor-based applications is necessary for the healthy operations of the overall system because the adversaries may target the proper functioning of applications and disturb the critical decision-making processes by injecting false information into the network. For instance, it is very important to provide authentic and accurate data to surrounding sensor nodes and to the sink to trigger time-critical responses [5]. There are two types of keying mechanisms for WSNs: static and dynamic. In static key management, the sensors are loaded with a fixed number of keys when the deployment of network is completed. In this key management schemes, key management functions (i.e., key generation and distribution) are handled statically. In the dynamic key management, schemes perform keying functions (rekeying) either periodically or on demand as needed by the network [1]. The sensors dynamically exchange the keys to communicate. The dynamic schemes are more attack resilient than the static ones, one significant disadvantage is the communication overhead will be increased because of keys which are being refreshed or redistributed from time to time in the sensor networks[3], [5]. The overhead associated with refreshing keys to avoid them becoming stale is to be minimized. Because from a sensors energy consumption point of view, the communication cost and the message transmission costs are very important issues in a Wireless Sensor network deployment[3].

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

II. BACKGROUND AND MOTIVATION

Sending confidential information from one node (source) to another node (destination) on a network could be a challenging task [5]. Using the available resources and energy, the nodes exchange data of the received and sent packets and also ensure data integrity before it hits the sink[1], [3] . The data exchanged could be manipulated or changed by the hacker on the network. So, the task would be to create a secure system that can ensure safety of the data using encryption methods and still use the available energy and resources without much overhead.
One significant aspect of confidentiality research in WSNs entails designing efficient key management schemes. This is because regardless of the encryption mechanism chosen for WSNs, the keys must be made available to the communicating node[4] . The keys could be distributed to the sensors before the network deployment or they could be distributed to nodes on demand as triggered by keying events. The former is static key management and the latter is dynamic key [2] management. In this work, we only consider dynamic Keying mechanisms in our analysis since OEMAC uses the dynamic keying paradigm. The main motivation behind OEMAC is that the communication cost is the most dominant factor in a sensors energy consumption [2], [5]. Thus, in this section, we present a simple analysis for the rekeying cost without the transmission of explicit control messages. According to the results of survey, the previous systems use the key hashing techniques which use more number of keys for encryption and decryption results in the consumption of more energy. As a result communication overhead increases. These key hashing techniques are less secure, which prone to more attacks [5].

III. EXISTING SYSTEM

An existing Dynamic Energy-based Encoding and Filtering framework to detect the injection of false data into a sensor network. Dynamic Energy-based that each sensed event report be encoded using a simple encoding scheme based on a keyed hash[4]. The key to the hashing function dynamically changes as a function of the transient energy of the sensor, thus requiring no need for re-keying. Depending on the cost of transmission vs. computational cost of encoding, it may be important to remove data as quickly as possible [3], [4]. Accordingly, DEEF can provide authentication at the edge of the network or authentication inside of the sensor network. Depending on the optimal configuration, as the report is forwarded, each node along the way verifies the correctness of the encoding probabilistically and drops those that are invalid. We have evaluated DEEF's feasibility and performance through analysis our results show that DEEF, without incurring transmission overhead [5].

IV. DRAWBACKS OF EXISTING SYSTEM

The insider attacks are outside the scope of this work. And no rekeying due to key revocation.

V. PROPOSED SYSTEM

OEMAC is a secure communication framework where the data is encoded using a scheme based on a permutation code generated via the RC6 encryption mechanism. The key to theRC6 encryption mechanism dynamically changes as a function of the residual energy of the network. Thus, a one-time dynamic key is employed for one packet only and different keys are used for the successive packets of the stream [3][3]. The intermediate nodes along the path to the sink are able to verify the authenticity and integrity of the incoming packets using a predicted value of the key generated by the senders virtual energy, thus requiring no need for specific rekeying messages [4].

VI. ADVANTAGES OF PROPOSED SYSTEM

1.OEMAC does not exchange control messages for key renewals as opposed to other dynamic key management schemes. Therefore, OEMAC is able to save more energy and is less chatty in nature.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

2. One key per message is employed. For the successive packets of the stream, different keys are used while the Previous schemes use basically the same key for different packets. This dynamic nature also makes the OEMAC framework more resilient to certain attacks (e.g., replayattacks, brute-force attacks, and masquerade attacks).
3. Since keys are generated in a separate module OEMAC, other key-based encryption or hashing schemes can also be adopted easily.

VII. SEMANTICS OF OEMAC

The OEMAC framework is comprised of three modules: Virtual Energy-Based Keying, Crypto, and Forwarding. The virtual energy-based keying process involves the creation of dynamic keys[3]. Contrary to other dynamic keying schemes, it does not exchange extra messages to establish keys. A sensor node computes keys based on its residual virtual energy of the sensor. The key is then fed into the crypto module. The crypto module in OEMAC employs a simple encoding process, which is essentially the process of permutation of the bits in the packet according to the dynamically created permutation code generated via RC6. The encoding is a simple encryption mechanism adopted for OEMAC. However, OEMACs flexible architecture allows for adoption of stronger encryption mechanisms in lieu of encoding .It consists of 2 modes.[3]

A. Operational mode: The OEMAC protocol provides three security services: Authentication, integrity and no repudiation. The fundamental idea behind providing these services is the watching mechanism. The watching Mechanism requires nodes to store one or more records (i.e. current energy level and Node-Id) to be able to compute the dynamic keys used by the source nodes, to decode packets, and to catch incorrect packets either due to communication Problems or potential attacks. However, there are costs (communication, computation, and storage) associated with providing these services. In reality, applications may have different security requirements. For instance, the security needed by a military application. Operational mode This is one of the operation modes in OEMAC. Here all nodes watch their neighbours, whenever a packet is received from a neighbour node, it is decoded and its authenticity and integrity are verified. Only valid or acceptable packets are forwarded toward the sink. In this mode, a short span of time exists at initial deployment so that no one can hack the network, because it takes time for an attacker to capture a node or get keys. During this period ,information to initialize route, may be used by each node to decide which node to watch and a record is stored for each of its one-hop neighbour in its watch-list. To obtain a neighbours initial energy value, a network-wise master key can be used to transmit this value during this period. Similar to the shared-key discovery phase of other dynamic key management schemes.

B. Statistical mode: In this operational mode, nodes in the network are configured to only watch some of the nodes in the network. Each node randomly picks node to monitor and stores the corresponding state before deployment. As a packet leaves the source node (originating node or forwarding node) it passes through node(s) that watch it based on probability. If the current node is not watching the node that generated the packet, the packet is forwarded. If the node that generated the packet is being watched by the current node, the packet is decoded and the plaintext ID is compared with the decoded ID. Similar to operational mode, if the watcher node wants to forward a packet and it cannot find the key successfully, it will try as many keys as the value of Key Search-threshold before actually classifying the packet as malicious. If the packet is authentic and the current hop is not the final destination then the original packet is forwarded, unless the current node is bridging the network. In the bridging case, the original packet is re encoded with the available bridge energy and forwarded. Since this node is bridging the network, both virtual and perceived energy values are decremented accordingly. If the packet is invalid or unacceptable, which is classified as such after exhausting all the virtual perceived energy values within the virtual Key Search Threshold window, the packet is discarded. This process continues until the packet reaches the sink. The forwarding module handles the process of sending or receiving of encoded packets along the path to the sink.

C. Keying Module: The virtual energy-based keying module of the OEMAC framework is one of the primary contributions of this paper. It is essentially the method used for handling the keying process. It produces a dynamic key

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

that is then fed into the crypto module [3]. The current value of the virtual energy, Evc, in the node is used as the key to the key generation function F. During the initial deployment, each sensor node will have the same energy level Eini, and therefore, the initial key, K1, is a function of the initial virtual energy value and an Initialization Vector (IV). The IVs are pre-distributed to the sensors. Subsequent keys, Kj, are a function of the current virtual energy, Evc, and the previous key is Kj-1. OEMACs virtual energy based keying module generates a new unique key as the data is sensed and ensures that each packet is associated with a new key generated [1], [3]. Generated dynamic key is passed to the crypt module. Each node computes and updates the transient value of its virtual energy after performing some actions. Each action (or state traversal) on a node is associated with a certain predetermined cost. The set of actions and their associated energies for OEMAC includes packet reception (Erx), packet transmission (Etx), packet encoding (Eenc), packet decoding (Edec) energies, and the energy required to keep a node alive in the idle state (Ea), transient value of the virtual energy, Ev, is computed by decrementing the total of these predefined associated costs, Ev, from the previous virtual energy value [3].

In order to successfully decode and authenticate a packet, a receiving node must keep track of the energy of the sending node. In OEMAC, the operation of tracking the energy of the sending node at the receiver is called watching and the energy value that is associated with the watched sensor is called Virtual Perceived Energy (Ep) as in [1]. Eg. node A starts with the value of 2,000mJ as the first key to encode the packet. Node A sends the first packet and decrements its virtual energy to 1,998 mJ. After node B receives this first packet, it uses the virtual perceived energy value as the key to decode the packet, and its Ep (1,998 mJ) after sending the packet[3].

Algorithm: Compute Dynamic Key
Compute Dynamic Key (Evc, ID)
begin j → txID
if j=1 then
kj → F(Eini, IV)
else
Kj → F(Kj−1,Evc)
end if
return Kj
end.

D. Crypto Module: The module performs simple encoding operation using the permutation of bits in the packet, according to the dynamically created permutation code via the RC6 encryption mechanism. The key to RC6 is created by virtual energy-based keying module. The purpose of the crypto module is to provide simple confidentiality of the packet header and payload while ensuring the authenticity and integrity of sensed data without incurring transmission overhead of traditional schemes[3], [5]. The packets in OEMAC consists of the ID (i-bits), type (t-bits) (assuming each node has a type identifier), and data (d-bits) fields. Each node sends these to its next hop. RC6 encryption algorithm takes the key and the packet fields (byte-by- byte) as inputs and produces the result as a permutation code. The concatenation of each bit output becomes the resultant permutation code [5]. Thus, instead of the traditional approach of sending the hash value. (e.g., message digests and message authentication codes) along with the information to be sent, we use the result of the permutation code value locally.

RC6 Algorithm: RC6 algorithm is a block cipher that was one of the finalists in the Advanced Encryption Standards (AES) competition. It is suitable for modification of its security strength because it has adjustable parameter (number of rounds) that directly affects its strength. The RC6 algorithm evolved from its predecessor RC5, a simple and "parameterized family of encryption algorithm"[3]. It consists of three components: a key expansion algorithm, an encryption algorithm, and a decryption algorithm. The parameterization is shown in the following specification: RC6-w/r/b, where w is the word size, r is the non-negative number of rounds, and b is the byte size of the encryption key.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

Encryption with RC-6 with w/r/b
input: plain text is stored in 4 w bits
input registers A, B, C, D
Number r of rounds
w-bit round keys S[0..... 2r+3]
Output: Ciphertext stored in A, B, C, D
Procedure: B=B+S[0]
D=D+S[1]
for {i = 1 to 20 do}
{
t = (B x (2B + 1)) <<< 5
u = (D x (2D + 1)) <<< 5
A = ((A t) <<< u) + S[2i]
C = ((C u) <<< t) + S[2i+ 1]
(A,B,C,D) = (B,C,D,A)
}
A = A + S[42]
C = C + S[43]
Input:
Ciphertext stored in four w-bit input
registers A,B,C,D
Number r of rounds
w-bit round keys S[0.....2r + 3]
Output:
Plaintext stored in A,B,C,D
Procedure:
C = C - S[2r + 3]
A = A - S[2r + 2]
for i = r downto 1 do
{
(A,B,C,D) = (D,A,B,C)
u = (D x(2D + 1)) <<< log2 w
t = (B x (2B + 1)) <<< log2 w
C = ((C - S[2i + 1]) >>> t) u
A = ((A - S[2i]) >>> u) t
}
D = D - S[1]
B = B - S[0]

E. Forwarding Module: This module is responsible for the sending of packets (reports) initiated at the current node (source node) or received packets from other sensors (forwarding nodes) along the path to the sink. The reports traverse the network through forwarding nodes and finally reach the terminating node, the sink.

VIII. EXPERIMENTAL RESULT AND ANALYSIS

In this mat lab tool is used .MATLAB is a high-level technical computing language and interactive environment for algorithm development, data visualization, data analysis, and numerical computation.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

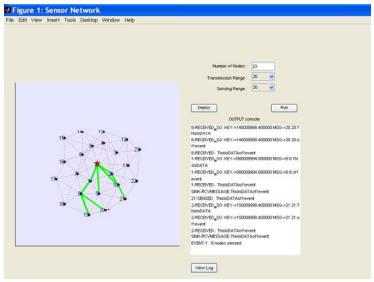

Figure 1. In the above figure an event occurred at the 20th node, nodes 1, 8,20, 19, 2 senses the event and sends encrypted data to the sink. This figure shows the deployment of nodes, sensing and sending encrypted data to sink.

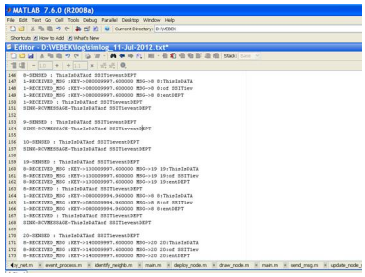

Figure 2. Nodes send data to sink with the decrease in virtual energy level which uses as key for the data transmission. It also shows each node having unique id.

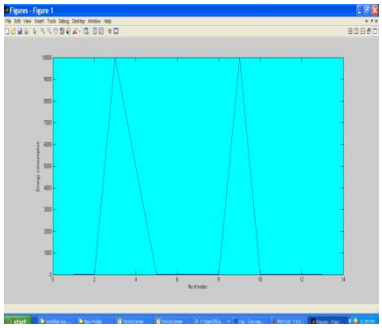

Fig. 3. In the above graph, x-axis gives the no of nodes and y-axis gives the energy consumed by the nodes. Here all the nodes have initial energy level10000. Nodes 1, 2, 5, 6, 7 , 8, 10, 11, 12,13 are involved in data transmission. Hence energy values of these nodes decreases to 9.9984, 9.9964 and so on. It depicts the consumption of energy by each node for the transmission.

ISSN(Online): 2319-8753
ISSN (Print): 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 6, Issue 7, July 2017

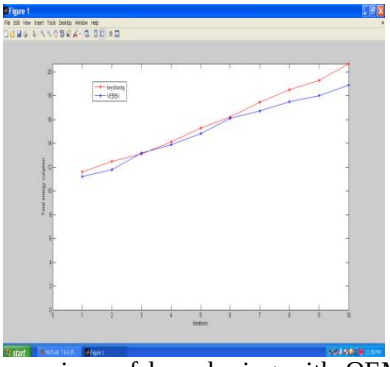

Figure 4. The above graph shows the comparison of key sharing with OEMAC. Here key sharing consumes more energy compared to OEMAC. It depicts comparison of OEMAC with Key sharing.

IX. CONCLUSION

Communication is very costly for wireless sensor networks and for certain WSN applications. Independent of the goal of saving energy, it may be very important to minimize the exchange of messages (e.g. military scenarios). To address these concerns, we presented a secure communication framework for WSNs called optimized energy management using advanced cryptographic algorithm. OEMAC uses RC6algorithm to perform encryption and keying. It is very easy to implement because it uses only primitive arithmetic operations on the blocks of plaintext.

REFERENCES

[1] Jing Deng, Yunghsiang S. Han, Shigang Chen, *A Key ManagementScheme for Wireless Sensor Networks Using Deployment Knowledge*,2004.
[2] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, *WirelessSensor Networks: A Survey*, *Computer Networks*, vol. 38, no. 4, pp. 393-422,March 2002.
[3] G.J. Pottie and W.J. Kaiser, *Wireless Integrated Network Sensors*, Comm.ACM, vol. 43, no. 5, pp. 51-58,2000.
[4] S. Uluagac, C. Lee, R. Beyah, and J. Copeland, *Designing SecureProtocols for Wireless Sensor Networks*, *Wireless Algorithms, Systems,and Applications*,vol. 5258, pp. 503-514, Springer,2008.
[5] Arif Selcuk Uluagac, *Virtual Energy-Based Encryption and Keying for Wireless Sensor Networks*, vol: IEEE transactions on mobile computing, vol. 9, no. 7, july 2010.