Available online at www.elixirpublishers.com (Elixir International Journal)

Adoc Network

Elixir Adoc Network 38 (2011) 4469-4472

ISSN: 2229-712X

Energy based encryption and keying with collision avoidance in wireless sensor networks

G. Ragunathan and L.Sherly Puspha Annabel

Department of MCA, St. Joseph's College of Engineering, Chennai-119.

ARTICLE INFO

Article history:

Received: 22 August 2011;

Received in revised form:

26 August 2011;

Accepted: 31 August 2011;

Keywords

Collision Avoidance,
Encryption,
Energy Based Keying,
Security.

ABSTRACT

Designing cost-efficient, secure network protocols for Wireless Sensor Networks (WSNs) is a challenging problem because sensors are resource-limited wireless devices. Resource limited in the sense limited energy level, memory and computational capacity. Since the communication cost is the most dominant factor in a sensor's energy consumption. We introduce an Energy-Efficient Energy Based Encryption and Keying with Collision Avoidance scheme for WSNs that significantly reduces the number of transmissions needed for rekeying. This is done to avoid stale keys. Energy based encryption is a secure communication framework where sensed data is encoded using a scheme based on a permutation code generated by using the RC4 encryption mechanism. Dynamic keys are used for the packets in the stream. The intermediate nodes in the path verify the authenticity and integrity of the incoming packets. The key generated by the sender's virtual energy, this avoids the rekeying. Energy based encryption efficiently transfers data between the nodes that are added in the watched list. Energy based encryption reduces transmission overhead. Transmission overhead increases if the packet size and also while transferring control messages for rekeying. During data transmission, if more than one node transmits data to same destination at same time the data gets collided at the receiving side. Retransmission reduces the energy level of the sensor node. While data is transmitted by first node and the next node must wait in queue until the first node completes its transmission. This mechanism avoids data collision in the destination and also it results in energy savings of a node because there is no need to transfer the same data again and again to the same destination if data collision occurs during transmission.

Introduction

There is a rapid development in WSN technology. WSN technology is used in variety of application such as military, environmental and also in commercial enterprises. To make use of the sensor applications in our daily lives there must be improvements in technology.

In the security point of view, it is very important to provide authentication and accurate data to nearby sensor nodes. Protocols must be effective to ensure the data transferred in the network is of accurate. Else it will result in utilization of resources. The tiny wireless devices are very large in number also it is deployed in the environments which is most probably unattended. Also these are limited in their capability and resources. These capabilities are said to be power, computational capacity, and memory.

This paper focus mainly on keying mechanism for WSNs. Static and dynamic are two fundamental key management schemes for WSNs. In static key management scheme functions are handled statically. But dynamic key management schemes perform keying rekeying functions either periodically or on demand when it is required by the network. Dynamic schemes are said to be flexible than that of the static ones. The dynamic key increases the communication overhead.

This paper also addresses the collision avoidance. If two nodes transmit data to the same destination, in the receiving side the data packets collision occurs frequently. Again the destination node requests the sender to retransmit the message. This process consumes the energy level of the node significantly and it results in the communication overhead. So before start transmitting the file the nodes ensure the destination is free to receive the data or some other node is transmitting data. Only is the destination is free, the node is allowed to transmit otherwise it waits until the first node completes data transmission.

Related Works

In sensor networks, the main concern is given to conserve the energy to maximize the life time of the network [2]. The messages from malicious nodes are filtered efficiently [6] and the integrity of the packets are maintained. To maintain the integrity, secret keys are used. There exist several key management schemes that are proposed for sensor networks. The main concern is to dynamically establish the security to the nodes. And also the maintaining the security is also given prime care. The key management features include energy awareness and local impact of attacks. That is the nodes in the network are limited to access their own data and not the data to other nodes.

The data is protected from the insider nodes in the path. The malicious nodes inject their own data [4] into the networks which results in consumption of the energy. Eliminate these data from the network to conserve the energy. This energy savings improves the life time of the nodes in the network. The dynamic key [6] management is more secure than that of the static ones.

The data transmission is done only once. Forwarding the same data to the destination again and again consumes the energy level significantly. So it must to ensure that data is

Tele: 9600307713

E-mail addresses: ragunathan_trichy@yahoo.co.in

transmitted without any collision. For that test whether the destination node is used by some other node to transmit data. If the destination is free start transmission or wait in the queue to start the transmission.

Energy-based encryption

It involves following operations.

- Node generation
- Key generation
- File transfer
- Watched list

Figure 1 explains the node details such as the node ID, energy level, transmission rate are maintained in the delegate side. This energy value, node ID and transmission rate is computed by the procedures. It is computed individually for each node. After computation it is stored in database. Then the link is established between the nodes. The neighbors are estimated and the cost to reach them is stored in the database.

After linking the nodes, the keying process is initiated. This energy based keying process involves the creation of dynamic keys. The dynamic key is generated with the details available in the node database. Also this methodology does not exchange extra messages to establish keys. A node computes keys based on its residual energy. These keys are generated by simple encoding process.

After the key is generated the file transfer operation transfers the file from source to destination. The node that wants to transmit the file must ensure that path is directly available to reach the destination. The data packets from source node reach the destination node with the help of the intermediate nodes [1]. In some paths there is no need of intermediate nodes to transmit the data from source to destination nodes. The path is estimated before transmitting the file. After path is estimated, the node ensures whether the destination is free to receive the data.

Then watched list framework eliminated the unwanted data [1]. After the node is added to the watched list it is allowed to transmit the message. The data injected by this node is matched with the list. If it matches it is considered as original data. Otherwise it is discarded in the transmission. By this way the security is given to the data. The generated key along with the data is transmitted in an efficient way.

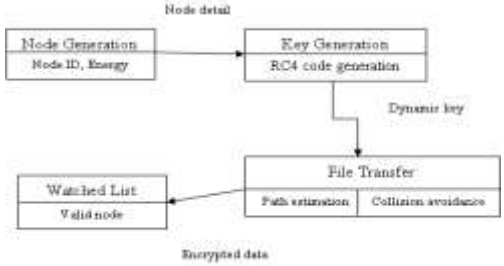

Fig. 1 Modular structure of Energy based encryption and collision avoidance framework

Node Generation

It is the framework meant for creation of the node details. It is computed by the method that maintains the node details such as the node id, energy level of each node. On the request of the node the function call will generate the node details. The node generation not only computes the details of the node as request by the node. It also computes the neighboring nodes of each node in the network. The hops between the nodes, the cost value to reach the neighboring nodes are estimated [3]. This is useful to estimate the path during file transmission. The path may contain intermediate nodes or no intermediate nodes. The cost value to reach the node and hops is fetched from database. The energy level is updated from time to time. The fluctuation in the battery levels of the nodes results in packet drop. So the energy is consumed significantly by retransmitting the messages again and again.

After deployment, nodes traverse several functional states. The states mainly include node-stay alive, packet reception, transmission, encoding and decoding. As each of these actions occurs, the energy in the node is depleted. Initially each node will have the same energy level. After the dynamic key is generated it is passed to key generation operation, where desired security services are implemented. So no mechanism is needed to refresh keys. Figure 2 explains how the energy value is absorbed during transmission

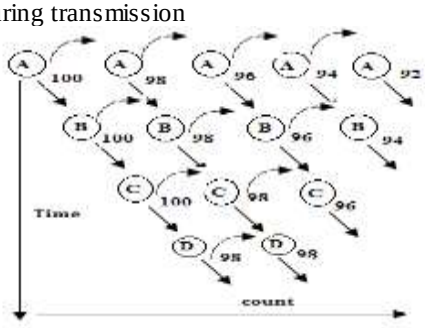

Fig. 2 Illustration of energy concept with forwarding

Key Generation

The dynamic key is generated for each node. The permutation code is generation operation ensures the security of the node. The traditional digital signatures or encryption mechanism requires expensive cryptography [5] is not possible. The encoding operation is essentially the process of permutation of the bits in the packet, according to the dynamically created permutation code via the RC4 encryption mechanism. The purpose 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. Key generation [6] and handling process is done in next framework. The packets consists of the ID (i-bits), type (t-bits) and data (d-bits) fields. Each node sends these to its next hop. Figure 3 explains how the RC4 encryption algorithm takes the key and the packet fields (byte by byte) as inputs and produces the result as a permutation code.

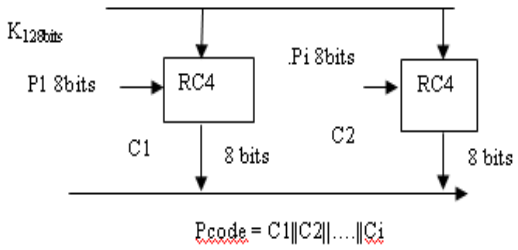

Fig. 3 An illustration of the use of RC4 encryption mechanism in energy based encryption.

Figure 4a shows the concatenation of each 8-bit output becomes the resultant permutation code. Figure 4b explains the resultant permutation code is used to encode the (ID|type|data) message. An additional copy of ID is also transmitted in the clear along with the encoded message. The format of final packet is = [ID, {ID, type, data}k]. Another significant step in the key generation operation involves how the permutation code dictates the details of the encoding and decoding operations over the fields of the packet when generated by a source sensor or received by a forwarder sensor. The permutation code P can be

mapped to a set of actions to be taken on the data stream combination. Figure 4c and 4d describes the permutation code computed by a node.

P = {1100100101}

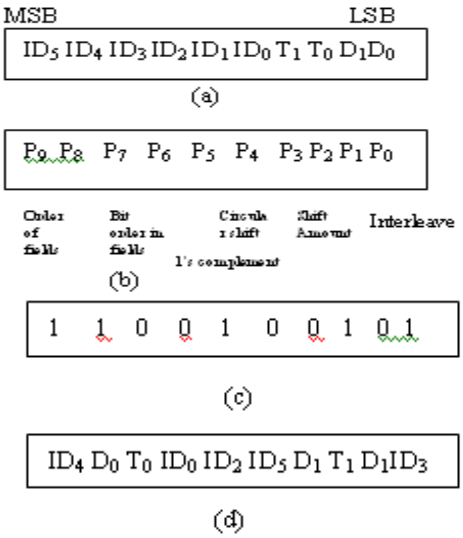

Fig. 4 Illustration of a sample encoding operation (a) i+t+d bit string before permutation. (b) Example encoding operations. (c) Example permutation code value. (d) i++d bit string after permutation.

The benefits of this simple encoding scheme are,

- Since there is no hash code to transmit, the packet size does not grow, avoiding bandwidth overhead. This increase the network lifetime.
- The technique is simple, thus ideal for devices with limited resources.
- The input to the RC4 encryption mechanism, namely, the key, changes dynamically without sending control messages to rekey.

File Transfer

The file transferring operation is the most important operation. Before transmitting the file the source node specify the message that it wants to transmit. Then the operation will estimate the path from source to destination. The neighbor node of each node is fetched from the database.

The path to the destination is with or without intermediate nodes. If there is no direct link between the two nodes then the path includes the intermediate nodes. So the initial stage of this operation computes the path estimation. Then the data collision is avoided. After the data is encrypted it is transmitted in the network.

Before starting the transmission the path is tested whether the destination is free to receive the message that it is going to transmit or the destination is used by some other node to transmit its message. For that the file is maintained. A node must test the flag value of the file. If the flag value is 0 then the node alters the value of the file to make other nodes wait in the queue until it finishes its transmission.

Then only the node is allowed to start the transmission. If another node wants to transmit the file while the first node operating, it finds that the path is used by some other. So it must wait until the first node completes transmission. By this way the data collision is avoided. It also results in the energy savings of the node. If the data is transmitted successfully then the retransmission is not necessary. By avoiding those retransmissions the energy level of the node is preserved significantly.

Watched List

Once the node receives the packet it will first check its watch-list to determine if the packet came from a node it is watching. If the node is not being watched by the current node, the packet is forwarded without modification or authentication It is done to maintain synchronization with nodes watching it further up the route.

If the node is being watched by the current node, the forwarding node checks the associated current energy stored for the sending node and extracts the energy value to derive the key. It then authenticates the message by decoding the message and comparing the plaintext node ID with the encoded node ID. If the packet is authentic, an updated energy is stored in the record associated with the sending node. If it is not authentic it is discarded. Again the energy value associated with the current sending node is only updated if this node has performed encoding on the packet.

To authenticate a packet, a node must keep track of the energy of the sending node to derive the key needed for decoding. Once the authenticating node has the initial energy value of the sending node, the value can be updated by decrementing the cost associated with the actions performed by the sending node using the cost equations defined in the previous sections on every successful packet reception.

Communication errors may be due to the deployment region while operating on unreliable protocols. Acknowledgement or data packets can be lost and the sender may not be able to determine which one actually was lost. Malicious nodes may also inject data. In such cases the energy value may differ.

The node that should have received the dropped packet and the nodes above the node on the path to the sink lose synchronization with the nodes below.

Subsequent packets generated, the next watching node will decode the packet with the virtual perceived energy key of the originating node and re encode the packet with the virtual bridge energy key, thus the network will be kept synchronized.

This framework was designed to avoid extra messages and not to increase the packet size to determine packet loss in network.

The next watching node tries to find the correct value of the perceived energy for the key within a window of energies. That is, the watching node decrements the predefined virtual energy value from the current perceived energy. When the node extracts key successfully, it records the newest perceived energy value and associates it with the sender node.

Operation Modes

There are 3 security services that are authentication, integrity and non repudiation. All nodes watch neighbors. If packet is received it is recorded and authenticated. Legal packets are only forwarded. If the key extraction is not successful it decrement predefined energy and tries another key. Re-encoding at every hop refreshes strength of encoding. It reduces transmission overhead because it eliminates malicious packet but increases processing overhead.

It also watches only some nodes in network. Each node randomly picks nodes to monitor. If it is not watched it is forwarded. If it is watched it is decoded & plain text is compared with decoded ID.

Conclusion

Communication is very costly for wireless sensor networks (WSNs) and for certain WSN applications. Independent of the

goal of saving energy, it may be very important to minimize the exchange of messages.

In comparison with other key management schemes, energy based encryption have the following benefits:

• It does not exchange control messages for key renewals and is therefore able to save more energy.
• It uses one key per message so successive packets of the stream use different keys-making the framework more flexible to certain attacks.
• It unbundles key generation from security services, providing a flexible modular architecture that allows for an easy adoption of different key based encryption or hashing schemes.
• Collision of data packets is avoided.

The energy performance of our framework with other en route malicious data filtering schemes saves energy of the sensor node. The result show that energy based encryption performs better than others while providing support for communication error handling, also the data collision is avoided which was not the focus of earlier studies. The future work will address insider threats and dynamic paths between the sensor nodes.

References

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless Sensor Networks: A survey", Computer Networks, vol. 38, no. 4, pp. 393-422, Mar. 2002.

[2] C. Vu, R. Beyah, and Y. Li, "A Composite Event Detection in Wireless Sensor Networks, "Proc. IEEE Int'l performance, Computing, and Comm. Conf. (IPCCC '07), Apr. 2007.

[3] Crossbow Technology, http://www.xbow.com, 2008.

[4] F. Ye, H. Luo, S. Lu, and L. Zhang, "Statistical En-Route Filtering of Injected False Data in Sensor Networks," IEEE J. Selected Areas in Comm., vol. 23, no. 4, pp. 839-850, Apr. 2005.

[5] R. Roman, C. Alcaraz, and J. Lopez, "A Survey of Cryptographic Primitives and Implementations for Hardware-Constrained Sensor Network Nodes," Mobile Networks and Applications, vol. 12, no.4, pp. 231-244, Aug.2007.

[6] H. Hour, C. Corbett, Y. Li, and R. Beyah, "Dynamic Energy-Based Encoding and Filtering in Sensor Networks", Proc. IEEE Military Comm. Conf. (MILCOM'07), Oct. 2007.