Efficient Key Establishment for Wireless Sensor Networks Using Elliptic Curve Diffie-Hellman

Tony Chung and Utz Roedig
Email: {{a.chung|u.roedig}@lancaster.ac.uk}

Infolab21, Lancaster University, UK

Abstract. We propose a broadcast method to establish symmetric keys between wireless sensor nodes and a sink, that achieves a different key for each node. We apply the Elliptic Curve Diffie-Hellman (ECDH) key exchange mechanism in two parts. The first part of the ECDH key exchange is conducted in a secure environment before network deployment to avoid the common man-in-the-middle problem of Diffie-Hellman (DH) schemes. The second part of the key exchange is initiated periodically by the sink using a broadcast message. Thus, the communication overheads in the resource constrained sensor network are reduced.

1 Introduction and Motivation

In many sensor networks, data readings are forwarded hop-by-hop towards a sink. Thereafter, the data is analysed at the central sink node. Such a sensor network scenario is considered within this paper.

Many wireless sensor network applications (for example physical intrusion detection systems) require end-to-end security between sensor nodes and the sink. The sink must be sure that the received data was originated by the sensor node and that it was not modified in transit. Thus, each sensor node and the sink must share cryptographic keys.

The used cryptographic keys need to be refreshed periodically to prevent cryptanalysis. Traditional key negotiation methods are not useful for the application domain of wireless sensor networks as they do not take their specific constraints on energy and communication bandwidth into account.

We propose a new way of using Elliptic Curve Diffie-Hellman (ECDH) to provide efficient and secure key exchange in wireless sensor networks. The first part of the ECDH key exchange is conducted before network deployment. The second part of the key exchange is initiated periodically by the sink using a broadcast message. As the first part of the exchange is executed in a secure environment, the man in the middle problem common to Diffie-Hellman scenarios is avoided. The use of a broadcast message to complete the second half of the key exchange allows us to minimise the communication overhead required to perform a key exchange.

2 Proposed Key Exchange Mechanism

Diffie-Hellman[1] is a well established method to agree a key k between two parties, A and B, without transmitting the key k over the insecure communication channel itself. For the proposed protocol we decided to use the Elliptic Curve (EC) variation of DH. EC cryptography provides the same security as classical cryptographic methods while using significantly shorter keys. Thus, the key material that has to be exchanged within DH is reduced, which is of importance in a resource constrained wireless sensor network.

Within elliptic curve DH (ECDH[2]), a key between A and B is established as follows. A and B agree a curve base G. A generates secret number a and calculates the corresponding public point $P = (x_a, y_a) = Ga$. B generates secret number b and calculates the public point $Q = (x_b, y_b) = Gb$. Then, P and Q are exchanged over the insecure channel. A and B can now calculate the shared key as $k = aQ = aGb = bP$. A possible attacker has only access to P and Q (and possibly G) which is not sufficient information to feasibly calculate k. However, an attacker might be able to intercept and modify all messages and, thus, negotiate a key k_a with A and k_b with B. The standard DH is prone to such man-in-the middle attacks.

The basic idea of the proposed ECDH based key exchange protocol is to use the previously described ECDH in the following way:

Phase 1 (Before Deployment)

1. All nodes are configured with the same EC parameters (including G).
2. a_n is generated for all n nodes and the $P_n = (x_n, y_n) = Ga_n$ are calculated.
3. a_n is stored on each corresponding node and all P_n are stored on the sink.

Phase 2 (After Deployment)

1. Regularly, the sink creates a new secret number b and public point $Q = Gb$.
2. The public point Q is then broadcast to all nodes.
3. Each node and the sink generate new keys $k_n = a_nQ = bP_n$.

3 Conclusion and Future Work

The proposed protocol has the benefit that unique keys can be set for all nodes in the network by sending just a single broadcast message. The key material can be included in broadcast messages that might be distributed by the sink to set other network parameters (e.g. routing messages for topology forming). We are currently implementing and evaluating the described protocol for TinyOS 2.x on TelosB motes.

References

1. New Directions in Cryptography. Whitfield Diffie, Martin E. Hellman. 1976. Stanford University.
2. A Public Key Infrastructure for Key Distribution in TinyOS Based on Elliptic Curve Cryptography. Malan, Welsh, Smith. Harvard University.