Chapter 10
Computationally Efficient Cooperative Public Key Authentication Protocols in Ubiquitous Sensor Network

Abedelaziz Mohaisen
University of Minnesota Twin Cities, USA

Tamer AbuHmed
Inha University,South Korea

DaeHun Nyang
Inha University, South Korea

ABSTRACT

The use of public key algorithms to sensor networks brings all merits of these algorithms to such networks: nodes do not need to encounter each other in advance in order to be able to communicate securely. However, this will not be possible unless "good" key management primitives that guarantee the functionality of these algorithms in the wireless sensor networks are provided. Among these primitives is public key authentication: before sensor nodes can use public keys of other nodes in the network to encrypt traffic to them, they need to make sure that the key provided for a particular node is authentic. In the near past, several researchers have addressed the problem and proposed solutions for it as well. In this chapter we review these solutions. We further discuss a new scheme which uses collaboration among sensor nodes for public key authentication. Unlike the existing solutions for public key authentication in sensor network, which demand a fixed, yet high amount of resources, the discussed work is dynamic; it meets a desirable security requirement at a given overhead constraints that need to be provided. It is scalable where the accuracy of the authentication and level of security are merely dependent upon the desirable level of resource consumption that the network operator wants to put into the authentication operation.

DOI: 10.4018/978-1-61350-110-8.ch010

INTRODUCTION

Public key cryptographic algorithms have been discarded from consideration as a solution for securing wireless sensor network (WSN) due to their long execution time (Chan et al., 2003). On the other hand, symmetric key algorithms have been intensively studied in the context of securing WSN due to their computational feasibility on the typical sensor nodes (Chan et al., 2003, Du et al., 2003, Eschenauer & Gligor, 2002, Liu & Ning, 2003, Perrig et al., 2002). However, recent results of operating public key algorithms on typical sensor nodes have shown a relevant and satisfactory efficiency. For example, Gura et al. in (Gura et al., 2004, Wander et al., 2005) introduced efficient implementation and measurements that show practicality of elliptic curve cryptography (Koblitz et al., 2000) and RSA (Rivest et al., 1983) signatures' verification; by showing that the ECC signature verification consumes 1.62 seconds on the 8-bit ATMega128 processor, which is the de facto standard processing unit in many commercialized sensor platforms (Crossbow Tech. Inc, BTnode Project). In addition, Watro et al. developed a limited public key architecture (called TinyPK) and provided an evaluation of practicality by measuring resources required per sensor to perform typical public key operations (i.e., signing, encrypting and decrypting) per sensor node in (Watro et al., 2004). The efficiency of key distribution in TinyOS based on ECC (Koblitz et al., 2000) is studied and measured on typical sensor nodes as shown by Malan et al. (Malan et al., 2004). All of these measurements, and recent studied, advocated the applicability of public key cryptography in the context of sensor network, and refuted the argument on the inefficiency of such algorithms for securing wireless sensor networks.

Indeed, public key algorithms have many advantages over the symmetric key algorithms, especially when deployed for WSNs. For example, while the resiliency to nodes compromise and connectivity of the sensor network security overlay are two critical issues in the latter type of algorithms, they are not a concern at all when using public key cryptography. This is, the compromise of a single node would reveal information related to that compromised node (its private key) and the ability of a node to encrypt a message to another node is subject to the knowledge of other nodes public key, making every node able to encrypt messages to arbitrary nodes in the sensor network without pre-existing knowledge of the destination. However, to make use of such algorithms, public key authentication is required. Worse, conventional public key authentication algorithms are inefficient for WSNs, for that WSNs are resources-constrained and such algorithms require more resources than that can be afforded by such networks.

Public key algorithms operate in a way that does not require a node to know other node's private key in order to encrypt a message intended for that node. However, before encrypting a message for the designated node, say Bob, the node wishing to encrypt the message, say Alice, needs to know Bob's public key in advance in order to be able to encrypt that message to him. Because of that, public key authentication is required to makes sure that Alice is encrypting for Bob using Bob's authentic public key. In the traditional networks, the public key authentication is performed using public key infrastructure and digital certificates, which are used for distributing public keys signed by the private of a centralized authority, acting as a trusted third party. Furthermore, when a node, Alice, wants to check the authenticity of the public key, she simply verifies the signature on the public key of Bob against the public key of the trusted authority, and thus realizes the authenticity of public keys. However, due to the special nature of the WSNs including, including the fact that the existence of such centralized authority in fully decentralized WSN is almost impossible, such solution is insufficient for these networks (Mohaisen