Securing Wireless Links on Resource-limited Devices

C. Kassapoglou-Faist

CSEM is completing its embedded security software library for wireless sensor networks, addressing entity authentication and key establishment. Solutions target mesh networks as well as point-to-point links and can be adapted to support standards (805.15.4-6, Bluetooth LE). This technology is being integrated in two CTI projects, on safety and industrial sensing applications.

Low-power wireless devices have become ubiquitous, ranging from simple remote sensors, to car keys, to health monitoring and embedded industrial-control devices, used in low-profile as well as in safety-critical applications. Nevertheless, the wireless medium is in essence an open communication channel—anyone in the vicinity can hear and produce messages. As a result, abnormal behavior, be it accidental or malicious, may lead to severe service malfunctions or even disruptions. There is an imperative need for appropriate security protection, tailored to the application requirements.

Adding to its expertise in wireless embedded systems, CSEM's objective is to acquire state-of-the-art practice in security solutions for wireless sensor networks (WSN) to build the corresponding software modules and to be able to propose adequate, thorough implementations that are easily integrated in an embedded communication protocol stack.

When two communicating devices establish a secure link (e.g. HTTPS), symmetric encryption is often used to protect the data, the key being established through an authenticated Diffie-Hellman (DH) exchange. This cannot be applied as such in the case of low-power wireless links, mainly due to the limited computing power and memory resources, driven by energy savings concerns and miniaturization requirements. Only elliptic curve DH can be considered, which still remains resource-demanding when compared to solutions based on symmetric cryptography. Moreover, the WSN communication model is often many-to-one (data collection) and one-to-many (commands from the sink), with possible pairwise links, notably for network organization and optimization purposes.

The proposed services are in line with WSN security requirements found in the literature:

- Message (or data) authentication, to guarantee that the data attributes are the ones claimed in the message. It also provides data integrity
- Confidentiality, which also enhances privacy protection
- Data freshness, for protection against replay of old valid messages (based on sequence numbers)
- Unilateral or mutual entity authentication for establishment of trust between the communicating parties, preventing man-in-the-middle attacks

In addition, access control and authorization must be implemented—in particular, the rejection of any non-expected message type or any non-expected behavior—to prevent node capture or system intrusion.

Cryptography is based on the Advanced Encryption Standard symmetric block cipher (AES-128). Data protection uses AES-CCM mode [1] (counter mode for encryption and cipher block chaining mode for data authentication). We have adopted the parameterizations taken in IEEE802.15.4 and Bluetooth LE.

The use of symmetric cryptography raises the issue of sharing secret keys. To make key compromise difficult, session keys, used to protect the messages, have time limited validity. They are derived from a master key, which is solely used for this purpose (and thus never exposed to cryptanalysis attacks), and from ephemeral data (nonces) exchanged by the devices in a mutual authentication and key establishment process, such as the one shown in Figure 1. Being already available, the AES function is also used in this phase (e.g. E(K,.) is AES, AES-CMAC as a hash function).

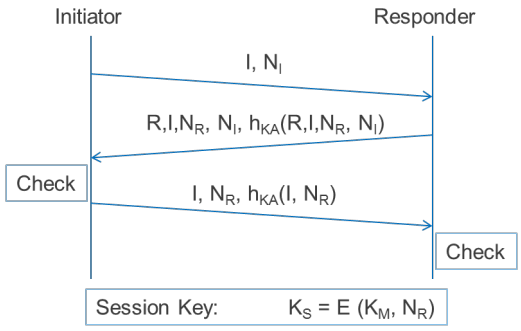

Figure 1: Typical mutual authentication scheme with symmetric key agreement (AKEP2).

There are several possibilities to distribute the master keys. They can be derived from pre-installed and/or configuration data or be established during a pairing procedure. For example, in Bluetooth LE, the parties first engage in a pairing procedure to establish a short-term key involving a process similar to Figure 1, then use this key to transport the long-term key (or master key) encrypted from one device to the other.

End-to-end data protection is proposed for WSNs: the messages are encrypted at their origin and decrypted/verified at final destination. Although the network level headers are thus transmitted in clear, all end-to-end data are authenticated. An individual, pairwise key per node-sink link is recommended for better robustness and easy key revocation. Depending on network topology, group keys are also needed. Either they are transmitted over the already secured sink-to-node link, or they are established between one-hop neighbors based on initial configuration data during network set-up, when an attack is considered very improbable.

Our security software runs on commercial (MSP430) as well as in-house (icyCOM) platforms. It is tested within WiseStack (the protocol stack around ULP medium access protocol WiseMAC). In addition, a BTLE implementation on an MSP430-icytrx platform successfully engages in pairing and exchange of encrypted data with an iOS i-pad.

[1] D. Whiting, *et al.*, "Counter with CBC-MAC" IETF RFC3610, (2003)