

Secure communication in networked embedded systems

M&C Cluster on Smart Buildings, 2.6.2010, Brussels

Markus Taumberger, VTT, Finland



© POBICOS Consortium 2010

POBICOS — Platform for Opportunistic Behaviour in Incompletely Specified, Heterogeneous Object Communities

Security considerations in WSNs

- Unique security threats because of
 - broadcast nature of the transmission medium
 - varying and often unattended deployment environments



Security threats

- **Eavesdropping** → Attack against privacy
- **Denial of service** → Disruption of the network
- **Message tampering** → Forwarding modified messages
- **Selective forwarding** → Malicious node drops messages
- **Sinkhole attacks** → Malicious node attracts all traffic of an area
- **Wormhole attacks** → Tunnelling messages through the network
- **Sybil attacks** → Malicious node masquerades multiple identities
- **Replay attacks** → Replaying recorded messages

Security requirements

- **Confidentiality** → Encryption to protect from eavesdroppers
- **Authentication** → Prevent injecting forged messages
- **Integrity** → Prevent the modification of messages
- **Freshness** → Protection against replay attacks

Key management

- **Single network-wide key**
 - All nodes share a single secret key
- **Pairwise shared keys**
 - Per-link key management
- **Hybrid-wise keys**
 - Using multiple types of keys simultaneously (base station, neighbour, cluster, network-wide)
- **Trusted server approach**
 - Trusted base station sets up session keys for nodes
- **Asymmetric cryptography**
 - Unique key-pair for each communicating entity
- **Random key pre-distribution**
 - Nodes get random subset of symmetric keys from large pool

Existing security architectures for WSNs

- **ZigBee security architecture with the ZigBee Trust Center**
 - Trust manager - Identifying and authenticating the devices that request to join the network
 - Network manager - Maintaining and distributing cryptographic keys to the devices
 - Configuration manager - Enabling end-to-end security between devices and binding two peer applications
- **SPINS - Suite of security protocols**
 - SNEP (Secure Network Encryption Protocol) - Provides confidentiality, twoparty authentication, integrity and freshness
 - TESLA - Provides authenticated streaming broadcast
- **TinySec**
 - TinySec-Auth - Only authentication is provided
 - TinySec-AE - Both authentication and confidentiality are provided
- **Public-key cryptography in WSNs**
 - Most security schemes for WSNs rely on symmetric-key cryptography
 - Promising experiments with the use of public-key cryptography in WSNs

Security and privacy in POBICOS



© POBICOS Consortium 2010

POBICOS — Platform for Opportunistic Behaviour in Incompletely Specified, Heterogeneous Object Communities

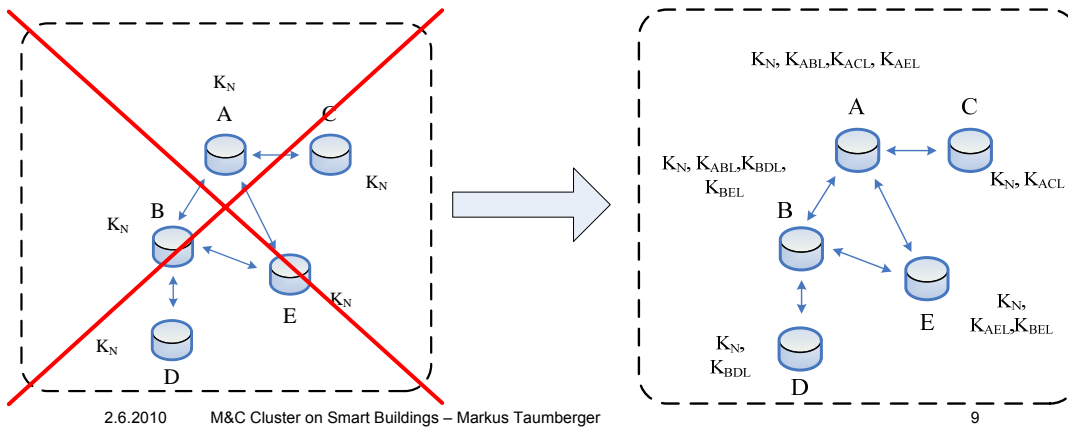
Background

- Prevention of eavesdropping and unauthorized system usage required
- Both SW and HW encryption utilized
- Two alternative security modes were specified:
 - One Network Key Approach
 - More lightweight solution, no protection against "insider attacks"
 - Identity-Based Cryptography Approach
 - Computationally more expensive, provides security for both "insider" and "outsider" attacks.
- Key management with Security Card Service running on e.g. user's PDA that utilizes close-proximity connection for the network key distribution.



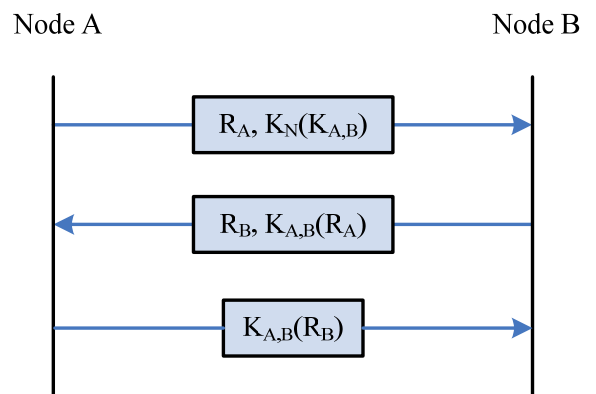
From one key to pairwise keys

- One network key mode turned out to be very tricky to implement securely.
 - Chosen encryption algorithm requires unique nonce value to be used for every message encrypted with the same key.
- An alternative mode was specified where each communicating node-pair shares a unique pairwise key.
 - Easy to ensure that nonce values don't reappear in the communications between two nodes.
 - More robust against eavesdroppers since the same key is not used all the time.
 - Allows for authentication within the network.



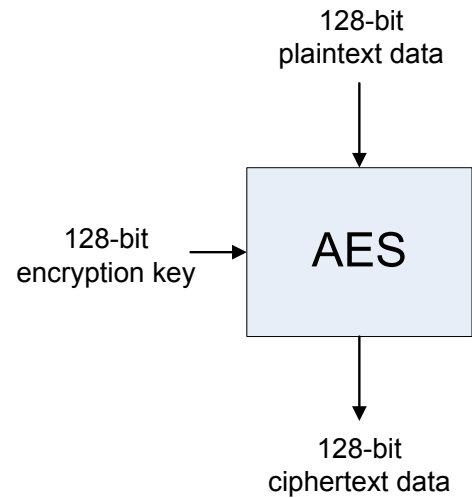
Pairwise-key establishment and authentication

- Pairwise key establishment cannot be performed by the user with the Security Card Service because new communicating pairs of nodes are formed at runtime.
- Solution: Security Card Service is used to distribute a network-wide key K_n that is used to exchange pairwise keys.
- Simple challenge-response protocol used for pairwise-key establishment and authentication.
- Note: Any node X within the network possessing K_n can obtain K_{AB} → trust must be placed on all devices that possess K_n .



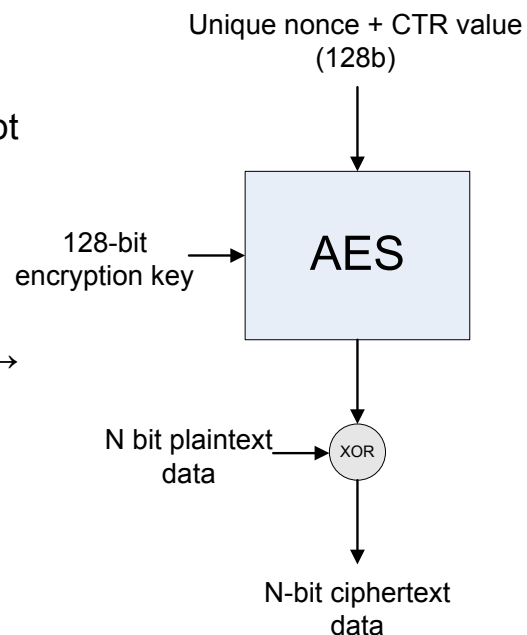
AES

- AES (Advanced Encryption Standard) block cipher algorithm for TinyOS.
 - Encrypts/decrypts blocks of 128 bits with a 128-bit key.
 - Optimized for 32-bit CPU such as the one on Imote2.
 - Initial speed tests with Imote2 running @104 MHz: 32bit: 62 us/block, 8bit: 268 us/block.
 - Not sufficient by itself to assure serious security e.g. there is no authentication and the same plaintext encrypted with same key gives always the same ciphertext.



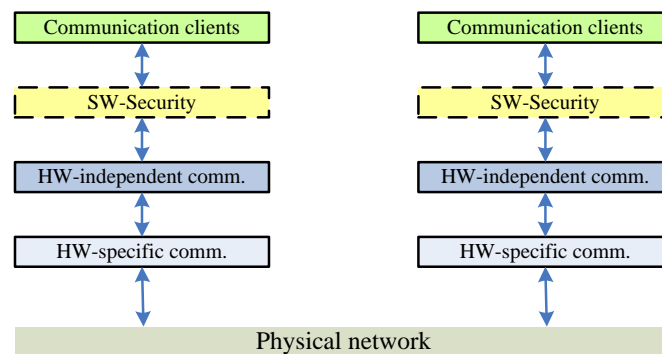
CCM

- CCM (Counter with CBC-MAC) mode of operation for AES.
 - Utilizes the AES block to efficiently encrypt and authenticate arbitrary length messages.
 - Adds MAC (Message Authentication Code) of 0-16B
 - Only AES encryption function is needed → faster operation
 - Unique nonce+CTR required for every message encrypted with the same key → problems with 1 network key solution.



SW Security module

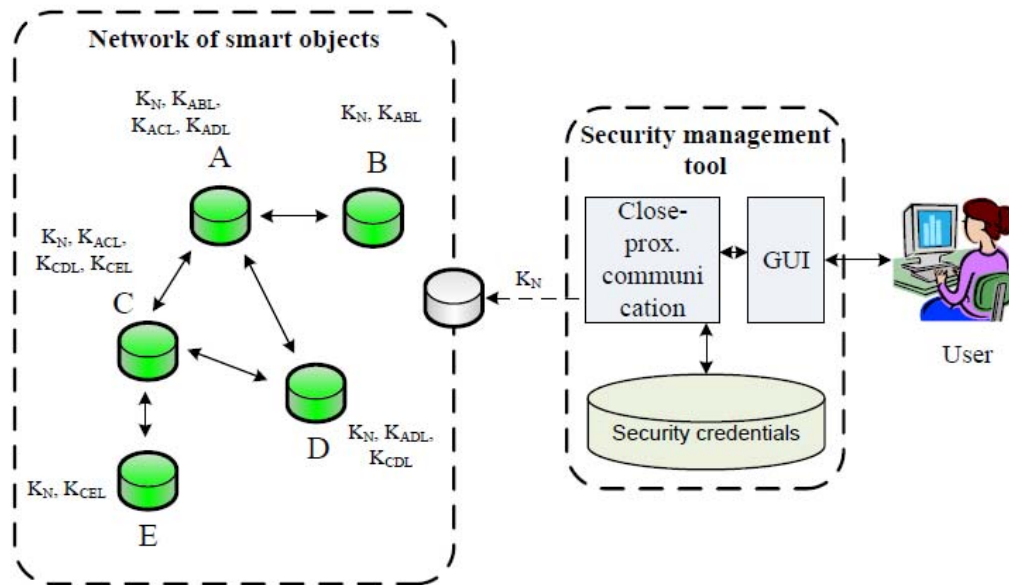
- Manages pairwise keys between nodes and encrypts/decrypts messages using CCM.
- Implements the protocol for pairwise key establishment.
- Operates independently on top of the communications module → SW security can be easily switched on/off without changes to existing design.



Close-proximity connectivity

- Close-proximity connection of the Security Card Service was implemented by using an Imote2 with reduced RF-power.
- Uses different radio chip and antenna than normal communications → no effect on normal message exchange.

Conceptual view of the security architecture



Future work

- Asymmetric cryptography
- Security zones
 - public
 - shared
 - private