SPINS: Security Protocols For Sensor Networks

Authors: Adrian Perrig, Robert Szewczyk, Victor Wen,
David Culler, J.D. Tygar
Presenter: Sujeet Mehta
Date: 04/28/03

1

Resource

- Adrian Perrig and Robert Szewczyk and Victor Wen and David Culler and J.D. Tygar, "SPINS: Security Protocols for Sensor Networks," in Proceedings of Seventh Annual International Conference on Mobile Computing and Networks, July 2001
- Available at: http://www.ece.cmu.edu/~adrian/projects/mc2001/mc2001.pdf

2

Outline

- Research Problem addressed by the paper
- Author's approach to tackle the problem
- Evaluation of the proposed techniques
- Related Work
- Future Work
- Conclusion

3

Research Problem

- Sensor networks: A special type of ad hoc network

  Large number of small sensors forming self-organizing wireless network

  One or more base station which interface sensor network to outside world

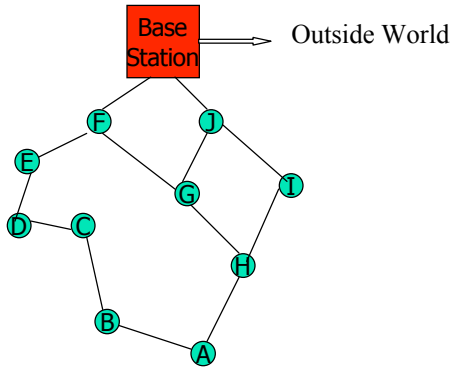
  Sensors collect the data and pass it to base station

  Base Station controls all the sensors

4

Research Problem cont'd

- Topology



5

Research Problem cont'd

- Sensor nodes are very cheap
  Limited Computational Capability
  Limited Storage and bandwidth
- Base Station is comparatively powerful
  Enough battery to surpass the lifetime of all sensor nodes
  Sufficient memory
  Means for communicating with outside world

6

Research problem cont'd

- Sensor networks to be used for emergency and life critical systems as well as military applications
- Hence, security is first and foremost. Need to provide data confidentiality, data authentication, data integrity and data freshness as in conventional desktop computers
- But…… new challenges
  - Cannot adopt same technique as used for conventional desktop computers
  - Main Reason: Sensors are cheap!
  - Sensors cannot perform public key operations (expensive)
  - Even with secret key operations extremely low overhead is required
  - Making it impractical to use majority of current secure algorithms
- Hence Research

7

Contribution of this paper

- Designing and developing two protocols for security in sensor networks
  - SNEP (Secure Network Encryption protocol)

    Provides Data confidentiality, two party data authentication and data freshness with low overhead
  - µTesla (micro version of Tesla)

    Providing authenticated broadcast

8

System Assumptions

- Individual sensors are untrusted
- Basic wireless communication is not secure
- Compromising the base station can render entire sensor network useless
- Each node trusts itself
- Before deployment each node is given a master key which is shared with the base station (all other keys are derived from this key)
- The communication pattern fall into three categories
  - Node to base station communication e.g. sensor readings
  - Base station to node communication e.g. specific requests
  - Base station to all nodes, e.g. routing beacons, queries or reprogramming of the entire network

9

Notation

A, B are principals, such as communicating nodes

N_A is a nonce generated by A (a nonce is an unpredictable bit string, usually used to achieve freshness).

$M_1 \mid M_2$ denotes the concatenation of messages M_1 and M_2

K_{AB} denotes the secret (symmetric) key which is shared between A and B

$\{M\}_{K_{AB}}$ is the encryption of message M with the symmetric key shared by A and B.

$\{M\}_{\langle K_{AB}, IV\rangle}$ denotes the encryption of message M, with key K_{AB}, and the initialization vector IV which is used in encryption modes such as cipher-block chaining (CBC), output feedback mode (OFB), or counter mode (CTR) [9, 21, 22].

10

Point to remember

- A master key shared between each sensor node and the base station before deployment
- From this master key, other keys are derived as follows

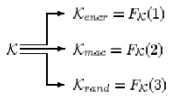


Figure 5: Deriving internal keys from the master secret key

11

SNEP: Data Confidentiality, Authentication, Integrity and Freshness

- Two party protocol
- Either for a base station to communicate with particular sensor node or for a sensor node to communicate with base station
- Also guarantees that two same plaintext messages do not produce two same ciphertext messages with the help of the counter 'C' shared between a sensor node and base station

$$A \rightarrow B : \{D\}_{\langle K_{encr}, C \rangle}, \mathrm{MAC}(K_{mac}, C \| \{D\}_{\langle K_{encr}, C \rangle})$$

12

Good Properties of SNEP

- Semantic Security
- Data Authentication
- Replay Protection
- Weak Freshness: If the message verifies correctly, a receiver knows that the message must have been sent after the previous message it received correctly
- Low Communication Overhead: The counter state is kept at each end point and does not need to be sent in each message

13

Strong Fairness in SNEP:

- Strong Fairness: An assurance to A that the message produced by B is in response to a specific message produced by A

$$A \rightarrow B : N_A, R_A$$
$$B \rightarrow A : \{R_B\}_{\langle K_{encr}, C \rangle}, \mathrm{MAC}(K_{mac}, N_A | C | \{R_B\}_{\langle K_{encr}, C \rangle})$$

14

µTesla: Authenticated Broadcast

- Base station wants to broadcast authenticated information to all the sensor nodes
- Point to point authentication scheme does not work fine for broadcast authentication
- Some sort of asymmetric algorithm is required, but impractical for sensor networks since sensor nodes are cheap
- Tesla protocol provides efficient authenticated broadcast but cannot be used directly for sensor networks
  - Tesla authenticates initial packet with a digital signature
  - Disclosing a key in each packet requires too much energy for sending and receiving
  - Expensive to store a one way key chain in a sensor node

15

µTesla: Authenticated Broadcast cont'd

- µTesla: A variant of Tesla
- It requires that the base station and nodes are loosely time synchronized, and each node knows an upper bound on the maximum synchronization error
- µTesla has multiple phases
  - Sender Setup:
    - Sender generates a sequence of secret keys i.e. $K_J = H(K_{J+1})$

16

µTesla: Authenticated Broadcast cont'd

- Broadcasting authenticated packets:
  - Time is divided into time intervals
  - Sender associates each key of one way key chain with each time interval
  - Sender will reveal the key K_T after a delay of _ intervals after the time interval T

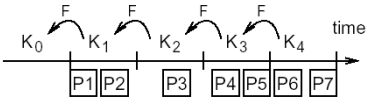


Figure 2: Using a time-released key chain for source authentication.

µTesla: Authenticated Broadcast cont'd

- Bootstrapping a new receiver
  - To authenticate sender's initial commitment of one way key chain to all its receiver

$$M \rightarrow S : N_M$$
$$S \rightarrow M : T_S \mid K_i \mid T_i \mid T_{int} \mid \delta$$
$$\text{MAC}(K_{MS}, N_M \mid T_S \mid K_i \mid T_i \mid T_{int} \mid \delta)$$

- Authenticating Broadcast Packets
  - Check the security condition i.e. check whether the key has already been revealed or not
  - If true buffer the packet else discard
  - As soon as node receives the key of a previous time interval, check for its authenticity
  - If check is successful, receiver can authenticate all the packets in that time interval

µTesla: Authenticated Broadcast cont'd

- Sensor nodes broadcast of authenticated data:
  - The node broadcasts the data through the base station
  - The node broadcasts the data however the base station keeps the one way key chain and send keys to the broadcasting node as needed

19

Implementation

- Sensor nodes
  - 8 KB of ROM
  - 512 bytes of RAM
  - The program memory used for TinyOS
- Block Cipher
  - Encryption algorithm is subset of RC5 from OpenSSL
- Encryption Function
  - Counter Mode is used
  - Stream cipher in nature
  - Offers semantic security and data freshness

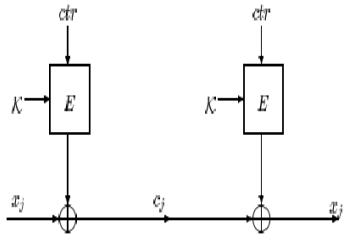


Figure 3: Counter mode encryption and decryption. The encryption function is applied to a monotonically increasing counter to generate a one time pad. This pad is then XORed with the plaintext. The decryption operation is identical.

20

Implementation cont'd

- Random Number Generation
  - MAC function used as a pseudo random number generator
  - Maintain a counter C that is incremented after each pseudo random block is generated i.e. Cth Pseudo Random Output: MAC (K_{Rand}, C)
- Message Authentication

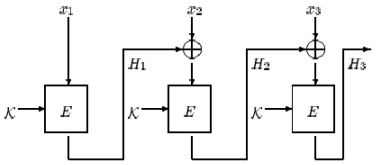


Figure 4: CBC MAC. The output of the last stage serves as the authentication code.

Evaluation

- Together crypto library and protocol implementation consume about 2KBytes of program memory, which is quite acceptable in most applications
- Performance
  - The amount of work needed for µTesla easily performed by sensor nodes
  - The performance of the cryptographic primitives is adequate for the bandwidth supported by current generation of sensor networks
- Energy Costs

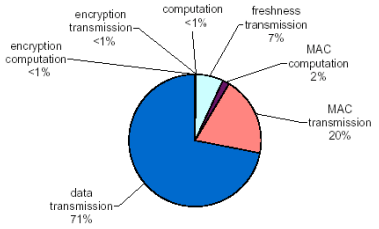


Figure 6: Energy costs of adding security protocols to the sensor network. Most of the overhead arises from the transmission of extra data rather than from any computational costs.

Applications: Authenticated Routing

- Assumption: Network consists of bidirectional communication channel
- The route discovery depends on periodic broadcast of beacon
- Route packet (beacon) originated at base station and authenticated using μTesla
- Node upon reception of beacon, checks whether it has already received a beacon in the current epoch
  - If it is the first one in that epoch, we accept it as the parent otherwise discard it

23

Application: Node to node Key Agreement

$$A \rightarrow B : N_A, A$$
$$B \rightarrow S : N_A, N_B, A, B, \mathrm{MAC}(K_{BS}, N_A | N_B | A | B)$$
$$S \rightarrow A : \{SK_{AB}\}_{K_{AS}}, \mathrm{MAC}(K'_{AS}, N_A | B | \{SK_{AB}\}_{K_{AS}})$$
$$S \rightarrow B : \{SK_{AB}\}_{K_{BS}}, \mathrm{MAC}(K'_{BS}, N_B | A | \{SK_{AB}\}_{K_{BS}})$$

24

Related Work

- Zhou and Hass propose to secure ad hoc networks using asymmetric cryptography

  L. Zhou and Z.J.Hass. Securing ad hoc networks. 13(6), November/December 1999
- Symmetric solution for broadcast authentication by Gennaro and Rohatgi

  R.Gennaro and P.Rohatgi. How to sign digital streams. In Burt Kaliski, editor, Advances in Cryptology
- Tesla Broadcast authentication scheme

  A. Perrig, R. Canetti, J.D. Tygar, D.Song. Efficient authentication and signing of multicast streams over lossy channels. In IEEE Symposium on Security and Privacy, May 2002
- Carman, Kruus and Matt analyze a wide variety of approaches for key agreement and key distribution in sensor networks

25

Future Work

- Efficient initial distribution of commitment of one way key chain for µTesla

  Donggang Liu, Peng Ning, "Multi-Level µTESLA: A Broadcast Authentication System for Distributed Sensor Networks," Technical Report, TR-2003-08, North Carolina State University, Department of Computer Science, March 2003.
- Information leakage through covert channels
- Focus on denial of service attack
- Focus on resource consumption attack
- Hardware advances in sensor nodes
- µTesla requires buffering at sensor nodes which are extremely constrained in terms of memory

26

Conclusion

- Demonstration of feasibility of implementing a security subsystem in limited sensor networks
- No use of public key cryptography
- Low overhead of additional security components
- Much more work needs to be done in this area and lot of research opportunities

27