Forward-secure Key Evolution in Wireless Sensor Networks

Marek Klonowski [1] Mirosław Kutyłowski [1]
Michał Ren [2] Katarzyna Rybarczyk [2]

[1] Wrocław University of Technology
Wrocław, Poland

[2] Adam Mickiewicz University
Poznań, Poland

CANS 2007, Singapore

Outline

What are sensor networks?

- Node capabilities:
  - sensing equipment
  - RF communication
  - processor
  - battery
- Network topology:
  - distances usually up to 30m apart
  - neighbors unknown in advance
  - 100–10000 nodes in a network

Dangers in sensor networks

- Nodes are not tamperproof, nor even tamper-resistant.
- Easy to steal node's keys with physical access.
- Vital to ensure that a single node compromise does not compromise the network
  - by eavesdropping
  - by inserting forged messages, including control messages
  - by making copies of a compromised node
  - etc.

Node capabilities

- Target price in cents per unit.
- Limited memory — too little to remember every key in the network.
- Limited CPU power — 8-bit processors, too slow for most public-key schemes.
- Symmetric encryption/decryption coprocessor.
- Limited energy.
- How to distribute keys to the nodes?

Random key predistribution

"A key-management scheme for distributed sensor networks" (2002), L. Eschenauer, V. D. Gligor[1]

- A large pool of keys is generated, and a random subset is loaded into each node.
- After deployment, nodes discover which keys they share with every neighbor. This establishes network topology — a "link" means that nodes share a common key.
- Nodes that are within RF range but do not share a key, must establish a path-key in order to communicate.
- Node compromise affects only a part of the network.

Key infection protocol

"Key Infection: Smart Trust for Smart Dust" (2004),
R. Anderson, H. Chan, A. Perrig[2]

- Every node creates its symmetric key randomly.
- Every node broadcasts its key in the clear.
- Assumption: the attacker is not omnipresent.
- Reasonable security if the attacker does not have his own sensor network already in place.

Key divergence scheme in a nutshell

"Diverging Keys in Wireless Sensor Networks" (2006), M. Ren, K. D. Tanmoy, J. Zhou[3]:

- Let's assume that nodes already share pairwise keys.
- When a node wants to communicate with another node, it transmits a message encrypted with a key that **differs by one bit** from the pairwise key.
- The other node has to crack the new key by brute-force — only a few tries are needed.
- When the other node succeeds and replies using the new key, the pairwise key is permanently changed.
- The procedure is repeated for as long as nodes communicate.

Energy savings

- One change of a bit in the divergence scheme adds $0.5 \mu J$ on average (for 128-bit keys).
- Transmitting 128 bits costs about $2500 \mu J$.
- Assuming that message structure does not make it necessary to make any extra transmissions (ECC, node identifiers and message counters are included in messages, and in replies), it is cheaper to change a key by divergence than by transmitting a new one.
- The energy gap between transmission and encryption will only widen in the future.

Attack scenario against KEP

- An adversary records wireless communication of a node.
- At a later time, the adversary captures the node, and extracts the key.
- Recorded communication together with the key allow the adversary to reverse KEP — reversing transitions requires finding only one flipped bit every time!
- All past communication becomes decrypted.

Forward-secure KEP

This paper:

- Forward security is desirable — subverting a node would not help in decrypting its past communication.
- Simple modification — instead of bit-flipping, let's use a one-way function.
- AES coprocessor can be used.

Forward-secure KEP in detail

Nodes A and B share a pairwise key: k_{AB}. Node A sends a message to B.

- A encrypts the message with $k' := F(k_{AB}, i)$, $i \in (1, \ldots, \ell)$, ℓ is a small constant, $\ell \geq 2$.
- From then on, A sends messages to B using k' until it receives a message from B.

Forward-secure KEP in detail — cont.

Node B receives a message from A.

- Message is encrypted with $k' := F(k_{AB}, i)$, $i \in (1, \ldots, \ell)$, i is not known to B.
- B discovers k' by trying all possible $i \in (1, \ldots, \ell)$.
- Next message sent by B to A will be encrypted with k'.

Forward-secure KEP in detail — cont.

Node A receives the message from B.

- Message is encrypted with k' which A chose, and B discovered.
- If the message is not fresh, indicating a replay attack, or if it is encrypted to a different key than k', then:
  - A rejects the message
  - A reverts to previous key k_{AB}
  - A remembers k' in case there was a communication error.
- Otherwise, A accepts k' as the new key.

Forward-secure KEP in detail — cont.

- From B's point of view, the exchange is as follows:
  - B receives a message from A encrypted with an unknown key k''.
  - B checks if $k'' = k_{AB}$.
  - If not, B checks every key of the form $F(k_{AB}, i)$, $i \in (1, \ldots, \ell)$.
  - In no key works, and B has unsuccessfully tried to change the key to k' earlier, it also checks all keys of the form $F(k', i)$, $i \in (1, \ldots, \ell)$.
  - If B succeeded in decrypting the message, and the message was fresh, B accepts k'' as the new key.
  - If the message could not be decrypted or was not fresh, it is rejected.

Potential problem

- Are all the keys reachable in the forward-secure KEP?
- Are the keys reachable quickly?
- Are all keys equally likely in the forward-secure KEP?
- Perhaps some keys are “attractors”, and the adversary could confine keyspace search to them?
- The previous case where keys changed by one bit was easy to analyze, the case with a one-way function is more difficult

Keyspace in Forward-secure Key Evolution

- K — set of possible keys
- $K = \{0, 1\}^n$, $N = |K| = 2^n$
- E — set of **directed edges**, such that for $k, k' \in K$, $kk' \in E \iff$ it is possible to change k into k' in one step of the protocol
- $G = (K, E)$ — graph representing the keyspace with transitions, in the key evolution protocol

Keyspace in Forward-secure Key Evolution — cont.

- One-way function F changes a key into one of ℓ keys, picked independently, uniformly at random
- Due to probability of a collision, the actual number of keys is in every step is a random variable X concentrated around ℓ
- We consider the random digraph $G(X) = (K, E)$, constructed by having each vertex v independently:
  - choose its out-degree ℓ_v according to the distribution of $X_v = X$
  - choose the set of ℓ_v out–neighbors uniformly from all ℓ_v-element subsets of K.

Strong connectivity

Our earlier question, rephrased: is $G(X)$ strongly connected?
If $E(X) = \ell \geq \ln N$, $\Pr(\lceil \frac{\ell}{2} \rceil \leq X) = 1$, $N \geq 2^{32}$ and $\ln N \leq \ell \leq \sqrt{N}/90 - 1$, then

Theorem (Strong connectivity)

With probability at least $1 - p'(N,\ell)$ $G(X)$ *is strongly connected, where:* $p'(N,\ell) = \frac{\ell}{N} \cdot \frac{N-\ell}{(N-2\ell)} e^{\left(\frac{2\ell(2\ell+1)}{N}\right)} + Ne^{\left(-\ell \cdot \frac{N-\ell-1}{N}\right)} + \frac{0.1(\ln N)^6}{N} + \frac{0.0017(\ln N)^{15}}{N^{1.99}} + \frac{1}{N^{0.59}} + \frac{1}{N^{0.16\ell}-1} + \frac{1}{N^{0.5}}$

Diameter

Our earlier question, rephrased: what is the diameter of $G(X)$?
If $E(X) = \ell \geq \ln N$, $\Pr(\lceil \frac{\ell}{2} \rceil \leq X) = 1$, $N \geq 2^{32}$ and
$\ln N \leq \ell \leq \sqrt{N}/90 - 1$, then

Theorem (Diameter)

With probability at least $1 - p(N)$*:*

$$\left\lfloor \frac{\ln N}{\ln 2\ell} \right\rfloor \quad \leq \quad \mathrm{diam}\, G(X) \quad \leq \quad \left\lceil \frac{\ln N}{2\ln\lfloor \frac{\ell}{2} \rfloor} \right\rceil + \left\lceil \frac{\ln N}{2\ln(\lceil \frac{\ell}{2} \rceil - 4)} \right\rceil + 4\,,$$

where: $p(N) = \frac{0.1(\ln N)^6}{N} + \frac{0.0017(\ln N)^{15}}{N^{1.99}} + \frac{1}{N^{0.59}} + \frac{1}{N^{0.16\ell - 1}} + \frac{1}{N^{0.5}}$

Results for typical keyspace sizes

N	ℓ	diameter	probability
2^{32}	32	$5-13$	$\approx 0,98$
2^{64}	64	$9-19$	$\approx 1-10^{-8}$
2^{128}	128	$16-26$	$\approx 1-10^{-17}$
2^{256}	256	$28-42$	$\approx 1-10^{-34}$

Let's denote the state of the keys after t steps as $P^t = (P_1^t, P_2^t \dots P_N^t)$, and assume that the transition function in every step of the KEP is randomly chosen. Our earlier question, rephrased: does any P_i^t deviate significantly from $1/N$?

Theorem (Deviation)

For step t, with parameters $N > \ell \geq 2$, for $\varepsilon > 0$, and $\delta = \frac{1}{\ell} - \frac{1}{N}$ we have:

$$\Pr\left(\max_i \left|P_i^t - \frac{1}{N}\right| \geq \varepsilon\right) \leq \left(\delta^t + \frac{\delta(1-\delta^{t-1})}{N(1-\delta)}\right)\varepsilon^{-2} .$$

Results for typical keyspace sizes

N	ℓ	t	probability	ε
2^{32}	32	32	0.001	$\approx 10^{-4}$
2^{64}	64	64	0.001	$\approx 10^{-9}$
2^{128}	128	128	0.001	$\approx 10^{-19}$
2^{256}	256	256	0.001	$\approx 10^{-39}$

Conclusion

- Forward security in the Key Evolution Protocol is feasible for keyspace sizes typical in sensor networks
- The KEP is able to provide several advantages for sensor network key distribution and management:
  - Compatibility with other key distribution schemes — can be added "on top".
  - Scalability — every node needs to remember only keys of neighbors.
  - Automatic increase in security (keys change faster) in high-traffic areas of the network.
  - Resistance to key compromise — pairwise keys are unique and change, so if a key is ever compromised, the attacker is forced to keep monitoring the connection, or lose the advantage.

Further reading

- L. Eschenauer and V. D. Gligor, “A key-management scheme for distributed sensor networks,” in *CCS ’02: Proceedings of the 9th ACM conference on Computer and communications security*, (New York, NY, USA), pp. 41–47, ACM Press, 2002.
- R. Anderson, H. Chan, and A. Perrig, “Key infection: Smart trust for smart dust,” in *Proceedings of IEEE International Conference on Network Protocols (ICNP 2004)*, Oct. 2004.
- M. Ren, K. D. Tanmoy, and J. Zhou, “Diverging keys in wireless sensor networks,” in *Information Security* (S. K. Katsikas, J. Lopez, M. Backes, S. Gritzalis, and B. Preenel, eds.), vol. 4176 of *LNCS*, pp. 257–269, Springer Verlag, 2006.
- M. Klonowski, M. Kutyłowski, M. Ren, and K. Rybarczyk, “Forward-secure key evolution protocol in wireless sensor networks,” *CANS 2007*, 2007.