

Key Distribution Protocols for WSN (probabilistic security)

Dan Cvrček
joint work with Petr Švenda

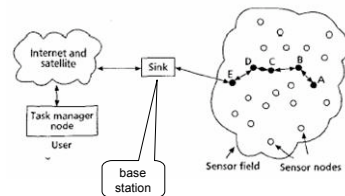
Overview

- Wireless Sensor Networks (WSN)
 - introduction
 - security goals, threads
- Key Distribution Protocols for WSN
 - Specifics of WSN environment
 - Common key distribution approaches
 - Randomised keys pre-distribution
- Plaintext key distribution (Key infection)

2/35

Wireless Sensor Networks

- Powerful base station(s) (BS)
- Network of nodes
 - sensing environmental properties
 - RF transceivers
 - battery powered
 - no tamper resistance
 - number of $10^3 - 10^6$
- Network topology
 - covering large areas
 - ad-hoc position/neighbours
 - distributed, multi-hop



3/35

Applications



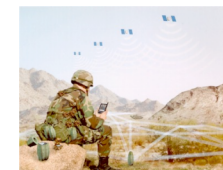
Traffic control



Medical monitoring



Wild fire detection

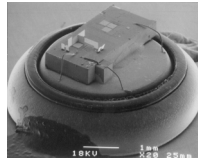
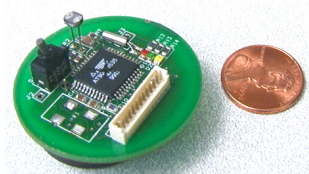


Battlefield management

4/35

Node hardware platform

- Berkeley Mote
 - 8-bit RISC processor
 - 4MHz clock
 - 512 B RAM
 - 8KB flash memory
 - OS code space: 3500 bytes
 - available code space: 4500 bytes
 - 10Kbps radio
- Berkeley's Smart Dust project
 - goal: node size < 1mm³
 - micro mirrors + laser beam
 - Micro-Electro-Mechanical Systems (MEMS)



5/35

Functional goals

- routing
 - QoS, load balancing, congestion reduction
- position discovery, localisation
- clustering
- query processing
- middleware
- ...

6/35

Security goals

- | | |
|---|---|
| <ul style="list-style-type: none">■ Secure routing■ Message security<ul style="list-style-type: none">□ confidentiality, integrity, authenticity■ Key & node revocation■ Network reinforcement<ul style="list-style-type: none">□ repeated deployment of new sensors■ Node authentication | <ul style="list-style-type: none">■ Resiliency<ul style="list-style-type: none">□ redundancy<ul style="list-style-type: none">■ battery, utility failure□ robustness<ul style="list-style-type: none">■ packet routing, active attack□ node capture<ul style="list-style-type: none">■ tolerant to % compromised■ perfect n.c. resilience - no other keys but captured get compromised |
|---|---|

7/35

Threats

- | | |
|---|---|
| <ul style="list-style-type: none">■ Eavesdropping■ Message injection■ Message modification■ Message replay■ Impersonation<ul style="list-style-type: none">□ clones | <ul style="list-style-type: none">■ DoS<ul style="list-style-type: none">□ Secure routing□ Malicious nodes□ Jamming□ Battery exhaustion■ Traffic analysis■ Side-channel analysis |
|---|---|

Possible difference from "Internet" systems
- data not sensitive in small numbers but as aggregates

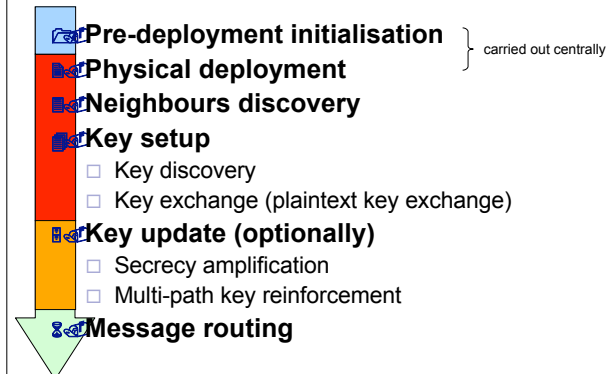
8/35

Key distribution protocols (KDP)

- Common KDP schemes inappropriate as WSNs have
 - restricted resources
 - limited neighbours/network topology knowledge
 - small (or none) tamper resistance
- Basic protocol requirements:
 - support for large number of parties
 - resource efficient
 - robust
 - single nodes compromise inevitable (no tamper resistance)
 - production defects, physical damage, battery exhaustion
 - (trusted) base-station (BS) involvement problematic
 - single point of failure
 - strong data flow around BS (non-uniform power exhaustion)

9/35

Bootstrapping protocol phases



10/35

Global master key

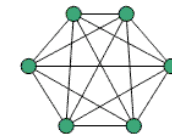
- Single symmetric key shared by all nodes
 - used for initial link key exchange and then erased
 - what if a node gets broken? (when being deployed, ...)
- Advantages
 - minimal storage requirements
 - resistance against DoS (fast MAC computation)
- Disadvantages
 - no node capture resilience
 - no nodes can be added later



11/35

Pair-wise keys ((n-1) scheme)

- Unique key between each two nodes
- Each node stores (n-1) keys, (n – size of network)
- Advantages
 - perfect resiliency to node capture
 - node-to-node authentication
- Disadvantages
 - high production costs
 - high memory requirements
 - network reinforcement impossible



12/35

Public key scheme

- Key pair for each node
 - signed by BS
- Advantages
 - perfect node capture resiliency
 - fully scalable, revocation possible
- Disadvantages
 - need for high performance hardware
 - high memory/time/power requirements
 - battery exhaustion attack possible
 - number of key establishment requests
- PK crypto doesn't bring much compared to symmetric one (works better in centralised environments)

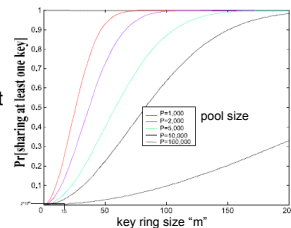
13/35

Probability schemes

14/35

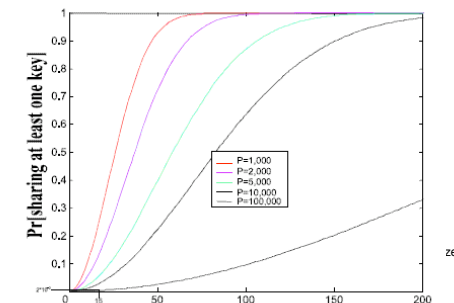
Random pre-distribution (EG)

- Idea (Eschenauer, Gligor - 2002):
 - two neighbours share pre-distributed key only with a certain probability p ($p \ll 1$)
 - basically, we need a connected network, not link keys
- Pre-deployment phase
 - large key pool size, each key with a unique ID
 - each node obtains random subset of m keys (no replacement)
- Key setup
 - neighbours use a common link key if such exists



15/35

Random pre-distribution (EG)

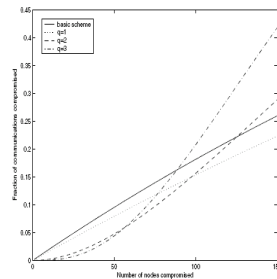


- Similar to birthday problem but "balls are drawn without replacement" => higher probability

16/35

Random pre-distribution (q-EG)

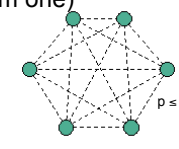
- Variation of the previous scheme (Chan, Perrig, Song)
 - (EG ~ 1-EG)
- $q \geq 1$ common keys required
 - $K' = \text{hash}(K_1 | \dots | K_q)$
- Node capture resilience should be improved **BUT**:
 - to keep link probability p same:
 - ring size m must be increased
 - pool size must be decreased
 - ...thus increasing # of compromised keys per node
- Search for function $(p, m, |S|)$ optimum



17/35

Random pairwise key scheme

- Idea (Chan, Perrig, Song - 2003):
 - two neighbours share pre-distributed pairwise key with $p \leq 1$
- Pre-deployment phase:
 - pairwise keys for m randomly chosen nodes
 - key between given two nodes is predetermined or not exists (not looking for random one)
- Properties:
 - perfect resilience to node capture
 - node-to-node authentication
 - limited network size ($n = m / p$)



5

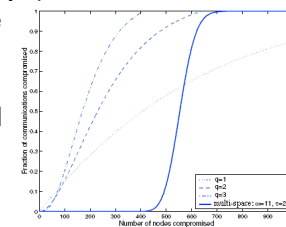
Single-space pairwise keys

- Blom's pairwise key pre-distribution schemes
- Each two nodes can compute unique pairwise key from their public and private values
- Less memory costing than $(n-1)$ scheme
 - $\lambda + 1$ elements ($\sim \lambda + 1$ keys)
- Perfectly secure until λ nodes captured
 - but totally compromised when $> \lambda$ captured
- Still inconvenient for WSN
 - linear dependency between memory and security

19/35

Multi-space pairwise keys

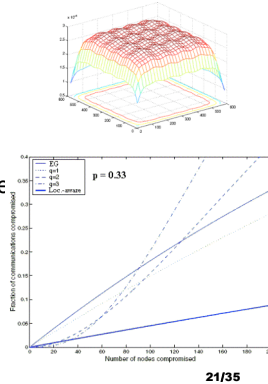
- (Du, Deng, Han, Varshney - 2003)
- Combination of Single-space + EG
 - key pool S contains Blom's key spaces
 - random subset for each node
 - pairwise key is constructed from shared Blom's space
- More resilient than EG until threshold reached



(a) $m = 200, p_{actual} = 0.333$

Location aware pre-deployment

- Limited location knowledge can be available (Du et.al.)
 - same deployment “barrel”...
- Deployment area grid
 - nodes from near “cells” are more probable to be communication neighbours
- One of previous schemes is performed “locally” for group of probable neighbour nodes



21/35

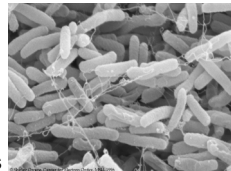
References

- Eschenauer, Gligor, *A Key-Management Scheme for Distributed Sensor Networks*, Computer and Communication Security, 2002
- Chan, Perrig, Song, *Random Key Predistribution Schemes for Sensor Networks*, IEEE Symp. Security Privacy, 2003
- Blom, *An Optimal Class of Symmetric Key Generation Systems*, EuroCRYPT84
- Du, Deng, Han, Varshney, *A pairwise key pre-distribution scheme for wireless sensor networks*, Computer and Communications Security, 2003 (extended TISSEC 2005)
- Du, Deng, Han, Chen, and Varshney, *A key management scheme for wireless sensor networks using deployment knowledge*, INFOCOM, 2004.
- Anderson, Chan, Perrig, *Key Infection: Smart Trust for Smart Dust*, Conference on Network Protocols, 2004

22/35

Key Infection - motivation

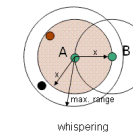
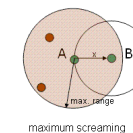
- More realistic attacker model
 - not able to eavesdrop the whole network
 - only a certain number of attacker's (black) nodes
- data from individual sensors is not sensitive
 - the real value is in aggregates
 - we don't we try to secure all nodes but “only” majority
- No keys are hardwired in nodes
 - low production costs
 - no danger in pre-deployment phases
- key material is distributed by ‘contact’, same way natural infection exploits



23/35

Key Infection - principle

- Restricted attacker's model
 - black/white ratio $\ll 1$
 - vulnerability period - shortly after deployment
- Plaintext key exchange with neighbours
 - keys established after deployment
 - after any network re-deployment
- Transmission modes
 - Maximum screaming
 - max. transmission power being used
 - Whispering
 - power is gradually being increased until a neighbour reached

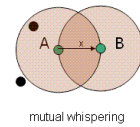


24/35

Secrecy amplification

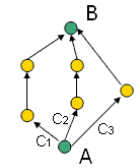
- Mutual whispering

- directional basic whispering
- $K = K_{AB} \oplus K_{BA}$



- Multipath key establishment

- key update sends values C_1, \dots, C_n along different paths
- $K' = K \oplus C_1 \oplus \dots \oplus C_n$
- attacker must eavesdrop all paths

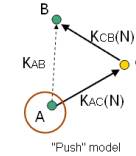


25/35

Multi-hop key establishment

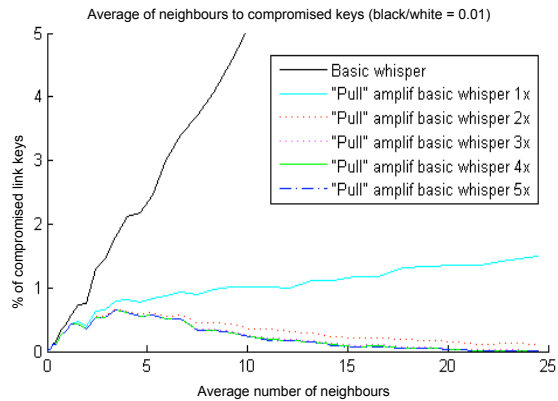
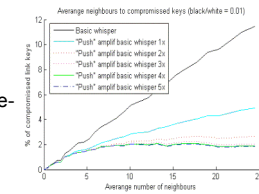
- Neighbours involved in key update

- 2-hop scheme: A, B participants, C mediator
- mediators immediately forget temporary values



- "Push" model (Anderson, Perrig, Chan)

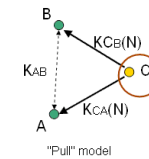
- initialised by participant A
- A asks mediator C to amplify K_{AB} by re-transmitting a number N
- $K'_{AB} = H(K_{AB} || N)$
- both A and B update link key to K'_{AB}



"Pull" model – our initial idea

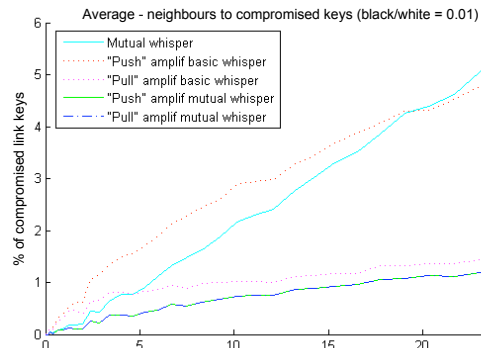
- "Pull" model

- initialised by mediator C
- C decides to amplify K_{AB} for A and B by sending N
- $K'_{AB} = H(K_{AB} || N)$
- both A and B update link key K_{AB}
- can be performed continually
 - ~ 3x amplification gives substantial improvements



28/35

Key infection - comparison

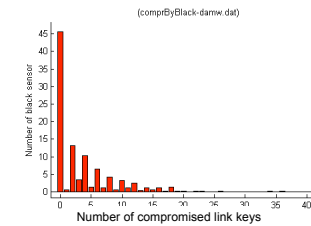


- (mutual + "Push") is equal to (mutual + "Pull")

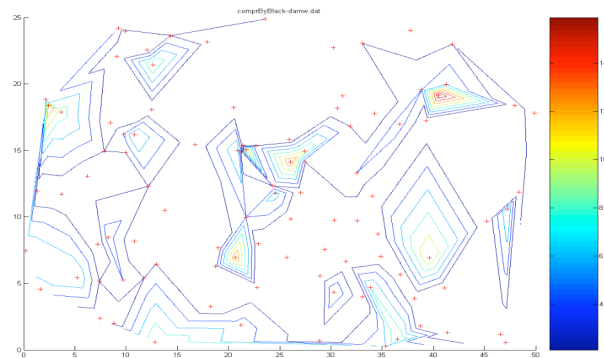
15

Key infection properties I

- Link keys are not compromised regularly
 - highly insecure areas
 - most areas are more secure than average



30/35



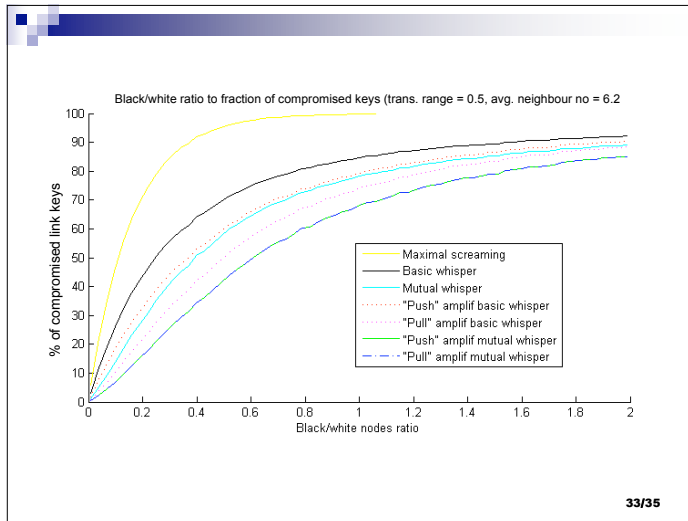
Compromised link key distribution (black/white = 0.01; avg no of neighbours = 6.2)
"Pull" amplification

31/35

Key infection – properties II

- Surprisingly many secure keys, even for high black/white nodes ratio
 - this deteriorates with increasing number of neighbours

32/35



A note

- Security is studied separately from e.g. routing problems and vice versa
- Routing protocols assume multiple neighbours while secure key exchange try to ensure at least one "connected" neighbour

34/35

Summary

- We realised that for avg no of neighbours of 6 => no of neighbours 0-20
- For dense enough network and moderate size of compromised fraction, no of secure link keys is very close to 100%
- There is a worst secrecy for a low no of neighbours
 - doesn't improve from a certain b/w ration depending on amplification method
- More questions than answers

35/35

