Secure Path-Key Revocation for Symmetric Key Pre-distribution Schemes in Sensor Networks

Tyler Moore and Jolyon Clulow

University of Cambridge
Computer Laboratory

22nd IFIP TC-11 International Information Security Conference
Sandton, South Africa

Outline

UNIVERSITY OF CAMBRIDGE

Ground rules for key management in sensor networks

- Sensor networks are comprised of low-cost, wireless devices
- Symmetric cryptography is preferred for computational efficiency
- Traditional key-exchange protocols are too expensive, so keys must be pre-distributed
- Sensors are cheap, so no tamper-proof hardware, and are deployed in unguarded areas
  - Threat model assumes a few nodes may be compromised to become active attackers
- Revoking the keys assigned to compromised nodes is essential

UNIVERSITY OF CAMBRIDGE

Options for pre-distributing keys

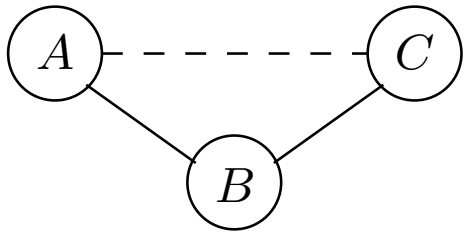
- Single master key pre-distribution
  - Inexpensive but susceptible to single compromise
- Pairwise key pre-distribution
  - Resilient to widespread compromise but storage infeasible for large networks (requires $n-1$ keys per node)
- Random key pre-distribution (Eschenauer & Gligor CCS 2002)
  - Nodes are assigned a random subset of keys from a large key pool
  - If nodes share a common key, then a link can be established
  - Probabilistic guarantees based on random graph theory
  - Efficient, though fails badly when a small group of nodes are compromised

Options for pre-distributing keys (ctd.)

- Random pairwise scheme (Chan *et al.* IEEE S&P 2003)
  - Combines the random graph approach with pairwise key assignment
  - More efficient than pure pairwise scheme, but requires much more storage than EG 2003 (each node typically stores between $0.2n$ and $0.4n$ keys, depending on parameters)
  - No duplicate keys, so secure against eavesdropping attacks
  - Pairwise key assignment enables mutual authentication between nodes sharing a key

UNIVERSITY OF CAMBRIDGE

Path-key establishment

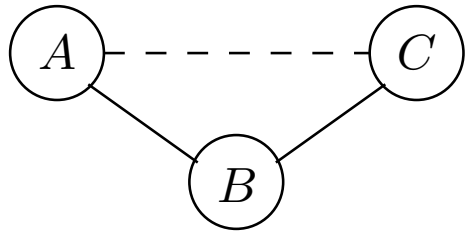


—— Pre-distributed key
– – Path key

- Whenever fewer than complete pairwise keys are pre-distributed, there must exist neighboring nodes not pre-assigned a key but wish to communicate
- One of the nodes chooses a new path key and sends it to the other node via intermediaries sharing keys

UNIVERSITY OF CAMBRIDGE

Path-key establishment (ctd.)



—— Pre-distributed key
– – Path key

- How are intermediate nodes chosen?
  - Random: nodes discover paths by asking neighbors about keys
  - Deterministic: link keys assigned based on identifier so nodes know who to ask
- Path-key setup is vulnerable to malicious intermediaries
  - Several papers propose ways to reinforce path keys by setting up keys using multiple disjoint paths
  - Fundamentally, there is no escaping the malicious intermediary problem

UNIVERSITY OF CAMBRIDGE

Threat model

- Attacker may actively compromise small minority of nodes $M_1, M_2, \ldots, M_i$
- Threat model **T.0**
  - Global passive adversary upon deployment
  - However, no nodes are actively compromised until path-key establishment is complete
- Threat model **T.1**
  - Global passive adversary upon deployment
  - A few nodes may be actively compromised prior to path-key establishment
  - Adopted by most key distribution schemes in literature

UNIVERSITY OF CAMBRIDGE

Revocation mechanisms

- Since threat models allow for the key material of several nodes to be compromised, revocation is an important step to minimize exposure and exclude further participation
- Centralized revocation scheme (Eschenauer and Gligor 2003)
  - Base station determines which keys are tied to a compromised node and instructs all nodes holding keys to delete them
- Distributed revocation schemes
  - Without a base station, no device has the authority to decide when a node should be removed or the keys to communicate a revocation instruction securely
  - Existing proposals let nodes vote to revoke each other (Chan et al. 2003, 2005) or unilaterally decide by imposing a cost (Moore et al. 2007)

Distributed revocation mechanism (Chan et al. 2005)

Voting Members

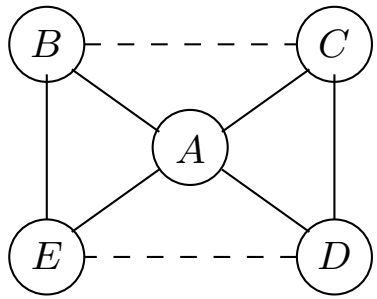
$V_A = \{B, C, D, E\}$

$V_B = \{A, E\}$

$V_C = \{ADE\}$

$V_D = \{A, C\}$

$V_D = \{A, B\}$



— Pre-distributed key

– – Path key

Stored Key Material

$A : \text{share}(\text{rev}_B), h^2(\text{rev}_B), \text{share}(\text{rev}_C), h^2(\text{rev}_C), \text{share}(\text{rev}_C), h^2(\text{rev}_C), \text{share}(\text{rev}_D), h^2(\text{rev}_D)$

$B : \text{share}(\text{rev}_A), h^2(\text{rev}_A), \text{share}(\text{rev}_E), h^2(\text{rev}_E)$

$C : \text{share}(\text{rev}_A), h^2(\text{rev}_A), \text{share}(\text{rev}_D), h^2(\text{rev}_D)$

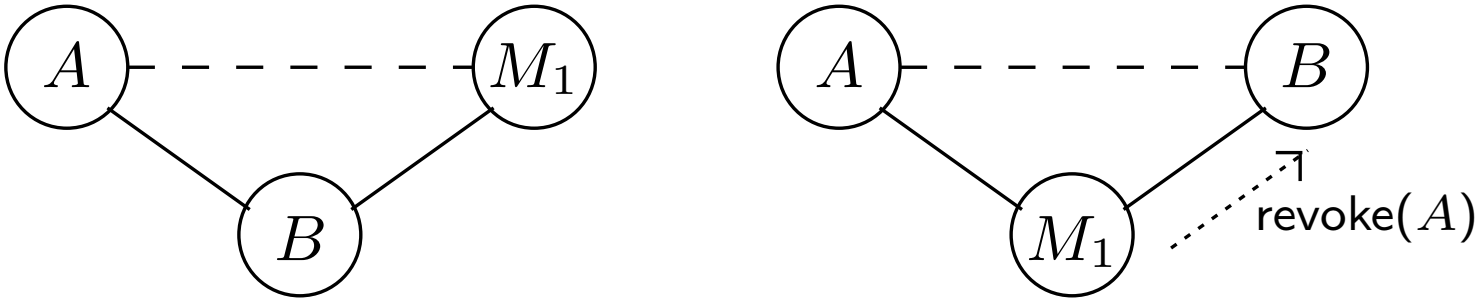
$D : \text{share}(\text{rev}_A), h^2(\text{rev}_A), \text{share}(\text{rev}_C), h^2(\text{rev}_C)$

$E : \text{share}(\text{rev}_A), h^2(\text{rev}_A), \text{share}(\text{rev}_B), h^2(\text{rev}_B)$

- Each node B that shares a pairwise key with A is assigned to the set of *participants of* A, V_A
- Each node A is assigned a revocation secret rev_A
- rev_A is divided into secret shares, given to all $B \in V_A$ and authenticator $h^2(\text{rev}_A)$
- Nodes vote against A by revealing their share
- If enough shares are revealed, rev_A is reconstructed and $h(\text{rev}_A)$ broadcast

UNIVERSITY OF CAMBRIDGE

Incomplete revocation of path keys



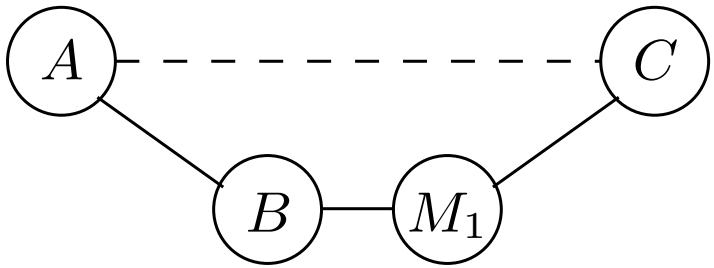
(a) Unrevoked path key (b) Spoofed revocation

—— Pre-distributed key – – Path key

- In Chan's distributed revocation scheme, only nodes that can verify votes are allowed to vote
- Only pre-assigned keys are revoked; no path keys established with revoked nodes are removed

UNIVERSITY OF CAMBRIDGE

Malicious intermediaries and path keys



— Pre-distributed key - - Path key

- The threat of malicious intermediaries interfering during path-key setup has been discussed in the literature
- What hasn't been considered is how malicious intermediaries can disrupt revocation mechanisms
- Any path keys exchanged via revoked nodes must also be revoked
- This matters for both threat models: even when no nodes are actively compromised during path-key setup, a global passive adversary can recover the path key later

UNIVERSITY OF CAMBRIDGE

More path-key attacks on revocation mechanisms

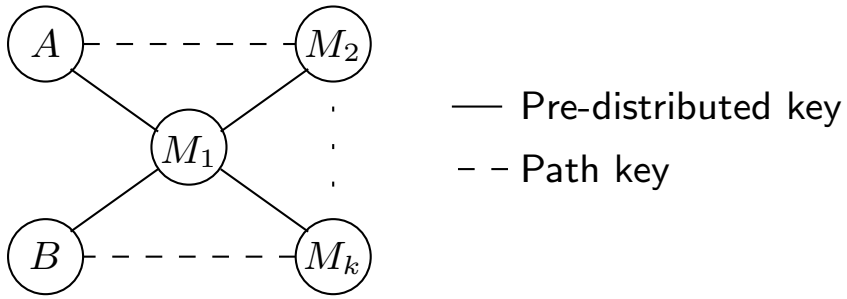
- Compromised but unrevoked pool keys
  - Eschenauer and Gligor's centralized revocation scheme advocates that nodes select unused pool keys as path keys
  - A malicious node can establish many path keys, requiring others to provide unused pool keys
  - Should the malicious node be revoked, it retains pool keys to get back into the network
  - Mitigating this by removing path keys enables a DoS attack
- Unauthorized reentry of revoked nodes
  - Two colluding malicious nodes can rejoin the network if only one of them is revoked
  - The revoked node sets up path keys via the remaining node
  - Works if honest nodes only delete keys and don't keep a blacklist

Sybil attacks in sensor networks

- In a Sybil attack, one malicious node pretends to be many distinct nodes
- Sybil attacks can disrupt routing, voting, data aggregation. . .
- In sensor networks, the keys possessed by a node are effectively its identity
  - Pool-key schemes are particularly susceptible to Sybils
  - Newsome et al. (2004) propose a Sybil-detection scheme where nodes challenge each other for unused pool keys
  - The authors claim that random-pairwise schemes are invulnerable to Sybils since keys aren't duplicated

UNIVERSITY OF CAMBRIDGE

Path-key-enabled Sybil attacks



- Path keys reintroduce the Sybil vulnerability to random-pairwise schemes
- A bad node can create many fake neighbors
- To counter this, a protocol is required that allows nodes not sharing keys to verify claims from nodes that do share keys

UNIVERSITY OF CAMBRIDGE

Countermeasures

- Complete notification of node revocation
- Path-key records to identify malicious intermediaries
- Blacklists to prevent unauthorized reentry via path keys
- We propose both centralized and decentralized mechanisms where appropriate

UNIVERSITY OF CAMBRIDGE

Complete notification of node revocation

- Every node that can establish a path key with a revoked node must be notified of its compromise
  - Where topology is unknown before deployment, every node must be notified
- Centralized revocation mechanisms can be augmented to unicast a message to every node instructing them to remove any path keys with the revoked node
- Decentralized revocation mechanisms
  - Nodes must be able to verify revocation messages sent by other nodes even when they don't trust them
  - Each node is loaded with authentication value for revocation secrets of all nodes
  - In Chan et al., nodes can only verify revocation if sharing pre-assigned keys

Path-key records to identify malicious intermediaries

- To prevent an adversary from collecting traffic & then recovering path keys following compromise (**T.0**), nodes update link keys via a one-way function
- What if a bad node is on the path when the key is exchanged (**T.1**)?
  - Store a path-key record with identifiers of nodes used
  - If node A sets up key with B, it stores

$$B, K_{AB}, N_1, N_2, \ldots, N_l$$

UNIVERSITY OF CAMBRIDGE

Path-key records (ctd.)

- Path-key record generation and verification should remain decentralized (if you can ask a base station, why not have it set up the path key for you?)
- Deterministic path-key establishment is required to keep bad intermediaries from tampering with the record
- Minimize the number of intermediaries used to reduce scope for tampering

UNIVERSITY OF CAMBRIDGE

Blacklists to prevent reentry via path keys

- Recall that under collusion, a previously revoked node can reenter by setting up path keys via an undetected bad node
- Centralized blacklist is undesirable since it would be necessary to consult each time a path key is exchanged
- Consistent, decentralized blacklist is simple to construct: nodes store identifiers of removed nodes whenever they see a revocation order

UNIVERSITY OF CAMBRIDGE

Cost summary

- Path keys impose the following additional costs for revocation to be effective:
  - Authenticated revocation secrets for all nodes in the network
  - Maintain path-key record listing all intermediaries on every path key
  - Maintain blacklist of all revoked nodes in network
- For symmetric key cryptography, these costs are unavoidable when anything less than complete pairwise keys are used
- Path keys also necessitate Sybil attack detection with indirect validation for pairwise key pre-distribution

UNIVERSITY OF CAMBRIDGE

Conclusions

- Pre-distributing less than complete pairwise symmetric keys weakens notions of identity – complications inevitably ensue
- We have shown how path keys undermine revocation mechanisms and facilitate Sybil identities
- We then proposed effective, albeit expensive, countermeasures
- Trade-offs made to improve the efficiency of bootsrapping keys ratchet up the costs of defending attacks during maintenance phases
- Open question: can lightweight uses of asymmetric crypto fare better than symmetric key schemes over the lifecycle of a sensor network?
- For more: `http://www.cl.cam.ac.uk/~twm29/`

UNIVERSITY OF CAMBRIDGE