# New Strategies for Revocation in Ad-Hoc Networks

Tyler Moore, Jolyon Clulow, Shishir Nagaraja and Ross Anderson

University of Cambridge
Computer Laboratory

Fourth European Workshop on Security and Privacy in Ad-Hoc and Sensor Networks (ESAS)
University of Cambridge, England

UNIVERSITY OF CAMBRIDGE

# Outline



UNIVERSITY OF CAMBRIDGE

# Outline



UNIVERSITY OF CAMBRIDGE

# Ground rules for key management in sensor networks

- Sensor networks are comprised of low-cost, wireless devices
- Computational efficiency is paramount, so symmetric cryptography is preferred (possibly supported by very limited asymmetric cryptography)
- Traditional key-exchange protocols are too expensive, so keys are often 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

# Threat model

- Attacker may actively compromise small minority of nodes
- Two threat models used in the literature
  - Conservative: global, active adversary upon deployment
  - Relaxed: adversary monitors at most a small fraction of communications during initialization
  - Model chosen affects the number of secrets that must be pre-loaded onto nodes
- Sybil attacks
  - In a Sybil attack, one malicious node pretends to be many distinct nodes
  - Node replication is a Sybil variant where copies of a subverted node are introduced
  - Sybil attacks can disrupt routing, voting, data aggregation...
  - We focus on networks where Sybil attacks can be contained

UNIVERSITY OF CAMBRIDGE

# Threat model

- Attacker may actively compromise small minority of nodes
- Two threat models used in the literature
  - Conservative: global, active adversary upon deployment
  - Relaxed: adversary monitors at most a small fraction of communications during initialization
  - Model chosen affects the number of secrets that must be pre-loaded onto nodes
- Sybil attacks
  - In a Sybil attack, one malicious node pretends to be many distinct nodes
  - Node replication is a Sybil variant where copies of a subverted node are introduced
  - Sybil attacks can disrupt routing, voting, data aggregation...
  - We focus on networks where Sybil attacks can be contained

UNIVERSITY OF CAMBRIDGE

# The problem of revocation in ad-hoc networks

- Three stages are required to revoke a bad node
  1. Detecting misbehavior
  2. Deciding when to revoke a node
  3. Implementing punishment
- Why are decision mechanisms hard to design properly?
  - Detection mechanisms rarely yield non-repudiable evidence (because signing every message is impractical)
  - More commonly, evidence is non-repudiable to a single party (e.g., MAC using pairwise key)
  - Repudiable evidence enables false accusations
  - Untrusted nodes are often better positioned to detect misbehavior than central authorities

UNIVERSITY OF CAMBRIDGE

# The problem of revocation in ad-hoc networks

- Three stages are required to revoke a bad node
  1. Detecting misbehavior
  2. Deciding when to revoke a node
  3. Implementing punishment
- Why are decision mechanisms hard to design properly?
  - Detection mechanisms rarely yield non-repudiable evidence (because signing every message is impractical)
  - More commonly, evidence is non-repudiable to a single party (e.g., MAC using pairwise key)
  - Repudiable evidence enables false accusations
  - Untrusted nodes are often better positioned to detect misbehavior than central authorities

UNIVERSITY OF CAMBRIDGE

# Existing decision mechanisms for sensor networks

- 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
  - Impractical unless a base station can detect misbehavior
- Distributed revocation schemes (Chan et al. 2003, 2005)
  - 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
  - Since detecting nodes cannot be trusted, then one logical response is to let devices vote for each other's removal

UNIVERSITY OF CAMBRIDGE

# Distributed revocation mechanism (Chan et al. 2005)

Voting Members

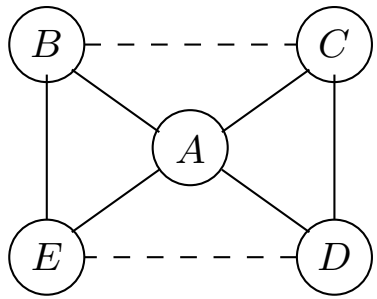
$V_A = \{B, C, D, E\}$

$V_B = \{A, E\}$

$V_C = \{ADE\}$

$V_D = \{A, C\}$

$V_D = \{A, B\}$




— Pre-distributed key

– – Path key

Stored Key Material

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

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

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

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

$E$ : share($\text{rev}_A$), $h^2(\text{rev}_A)$, 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 $A$*'s voting members*, $V_A$
- Each node $A$ is assigned a revocation secret $\text{rev}_A$
- $\text{rev}_A$ 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, $\text{rev}_A$ is reconstructed and $h(\text{rev}_A)$ broadcast

UNIVERSITY OF CAMBRIDGE

# Limitations to Chan's blackballing scheme

- No path keys are revoked
  - 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
  - Can be remedied by equipping nodes with authentication values for revocation secrets of all nodes
- Nodes cannot move around after deployment, otherwise a threshold of colluding bad nodes could roam around ejecting devices at will
- Fairly stringent computational, storage and communication requirements
- Delayed response for voting threshold to be reached

UNIVERSITY OF CAMBRIDGE

# Outline



UNIVERSITY OF CAMBRIDGE

# Reelection

- Chan's blackballing scheme utilizes negative votes – nodes condemn misbehavior
- We propose a system based on positive votes – good nodes periodically reelect each other to the club
- We discuss two variants of the reelection strategy
  - Reelection for semi-capable devices
  - Lightweight reelection using buddy lists

UNIVERSITY OF CAMBRIDGE

# Reelection for semi-capable devices

- Logical complement to secret-sharing-based blackballing
  - Each node $A$ must periodically present a network access token $\text{access}_{A,i}$ to remain on the network during time period $i \in \{1, \ldots, T\}$, created using a hash chain
  - Each token $\text{access}_{A,i}$ is divided into secret shares given to $A$'s voting members
  - $A$'s voting members cast votes by revealing their shares each period to reconstruct $\text{access}_{A,i}$
  - $\text{access}_{A,0}$ is distributed to each voting member to authenticate $\text{access}_{A,i}$
- Properties of secret-sharing-based reelection
  - To vote against $A$, a node must simply delete its shares
  - Votes are honored even if the node is later compromised
  - Storage costs are improved over blackballing because voting members do not need to prove to each other that a vote is valid

UNIVERSITY OF CAMBRIDGE

# Reelection for semi-capable devices

- Logical complement to secret-sharing-based blackballing
  - Each node $A$ must periodically present a network access token $\text{access}_{A,i}$ to remain on the network during time period $i \in \{1, \ldots, T\}$, created using a hash chain
  - Each token $\text{access}_{A,i}$ is divided into secret shares given to $A$'s voting members
  - $A$'s voting members cast votes by revealing their shares each period to reconstruct $\text{access}_{A,i}$
  - $\text{access}_{A,0}$ is distributed to each voting member to authenticate $\text{access}_{A,i}$
- Properties of secret-sharing-based reelection
  - To vote against $A$, a node must simply delete its shares
  - Votes are honored even if the node is later compromised
  - Storage costs are improved over blackballing because voting members do not need to prove to each other that a vote is valid

UNIVERSITY OF CAMBRIDGE

# Lightweight reelection using buddy lists

- Threshold secret-sharing-based blackballing and reelection remain relatively expensive: from reconstructing secrets to pre-assigning, swapping and storing shares
- Alternatively, nodes could transmit a buddy list of approved neighbors
  - Devices can cross-reference lists to check whether enough nodes have also approved their buddies
  - Buddy lists are approved using Guy-Fawkes style hash chains: nodes distribute key authentication values to neighbors upon deployment
- Advantages of buddy lists
  - No pre-assigned storage is required
  - Naturally supports diverse strategies towards risk

UNIVERSITY OF CAMBRIDGE

# Lightweight reelection using buddy lists

- Threshold secret-sharing-based blackballing and reelection remain relatively expensive: from reconstructing secrets to pre-assigning, swapping and storing shares
- Alternatively, nodes could transmit a buddy list of approved neighbors
  - Devices can cross-reference lists to check whether enough nodes have also approved their buddies
  - Buddy lists are approved using Guy-Fawkes style hash chains: nodes distribute key authentication values to neighbors upon deployment
- Advantages of buddy lists
  - No pre-assigned storage is required
  - Naturally supports diverse strategies towards risk

UNIVERSITY OF CAMBRIDGE

# Outline



UNIVERSITY OF CAMBRIDGE

# Suicide

- Any voting-based decision mechanism is necessarily complex and slow since many actors are involved
- Decisions are much simpler if a single device can decide
- Unfortunately, false accusations can undermine unilateral decisions
- Our solution: make punishment expensive
- Upon detecting misbehavior, a device commits suicide by broadcasting an instruction to remove the bad node and itself

UNIVERSITY OF CAMBRIDGE

# Implementing suicide

- Suicide using a central authority
  - Universally trusted base station can be leveraged to transmit authenticated suicide notes
  - If $A$ detects $M$ misbehaving, it sends $\text{suicide}_{A,M}$ to a base station, which verifies the message and sends out authenticated messages to other nodes
  - Notably, the decision remains distributed
- Distributed suicide using signatures
  - $A$ broadcasts a signed suicide note $\text{suicide}_{A,M}$
  - Other nodes verify the signature and delete keys shared with $A$,$M$
  - Can be implemented using public key crypto or one-time signatures

UNIVERSITY OF CAMBRIDGE

# Implementing suicide

- Suicide using a central authority
  - Universally trusted base station can be leveraged to transmit authenticated suicide notes
  - If $A$ detects $M$ misbehaving, it sends $\text{suicide}_{A,M}$ to a base station, which verifies the message and sends out authenticated messages to other nodes
  - Notably, the decision remains distributed
- Distributed suicide using signatures
  - $A$ broadcasts a signed suicide note $\text{suicide}_{A,M}$
  - Other nodes verify the signature and delete keys shared with $A$,$M$
  - Can be implemented using public key crypto or one-time signatures

UNIVERSITY OF CAMBRIDGE

# Challenges to distributed suicide

- Flypaper attacks
  - Bad node presents widely observable misbehavior to attract simultaneous suicides
  - Centralized scheme: base station can choose which offer to accept
  - Decentralized scheme: (i) randomized back-off before sending offer and (ii) as tie-breaker, accept the offer with earliest timestamp
- Trolling attacks
  - Bad node presents itself in several locations, either re-using identities (node replication) or presenting different ones (Sybil)
  - Need detection mechanisms for Sybil and node replication attacks
  - Multiple-offer resolution can identify reused identities if network is connected

UNIVERSITY OF CAMBRIDGE

# Challenges to distributed suicide

- Flypaper attacks
  - Bad node presents widely observable misbehavior to attract simultaneous suicides
  - Centralized scheme: base station can choose which offer to accept
  - Decentralized scheme: (i) randomized back-off before sending offer and (ii) as tie-breaker, accept the offer with earliest timestamp
- Trolling attacks
  - Bad node presents itself in several locations, either re-using identities (node replication) or presenting different ones (Sybil)
  - Need detection mechanisms for Sybil and node replication attacks
  - Multiple-offer resolution can identify reused identities if network is connected

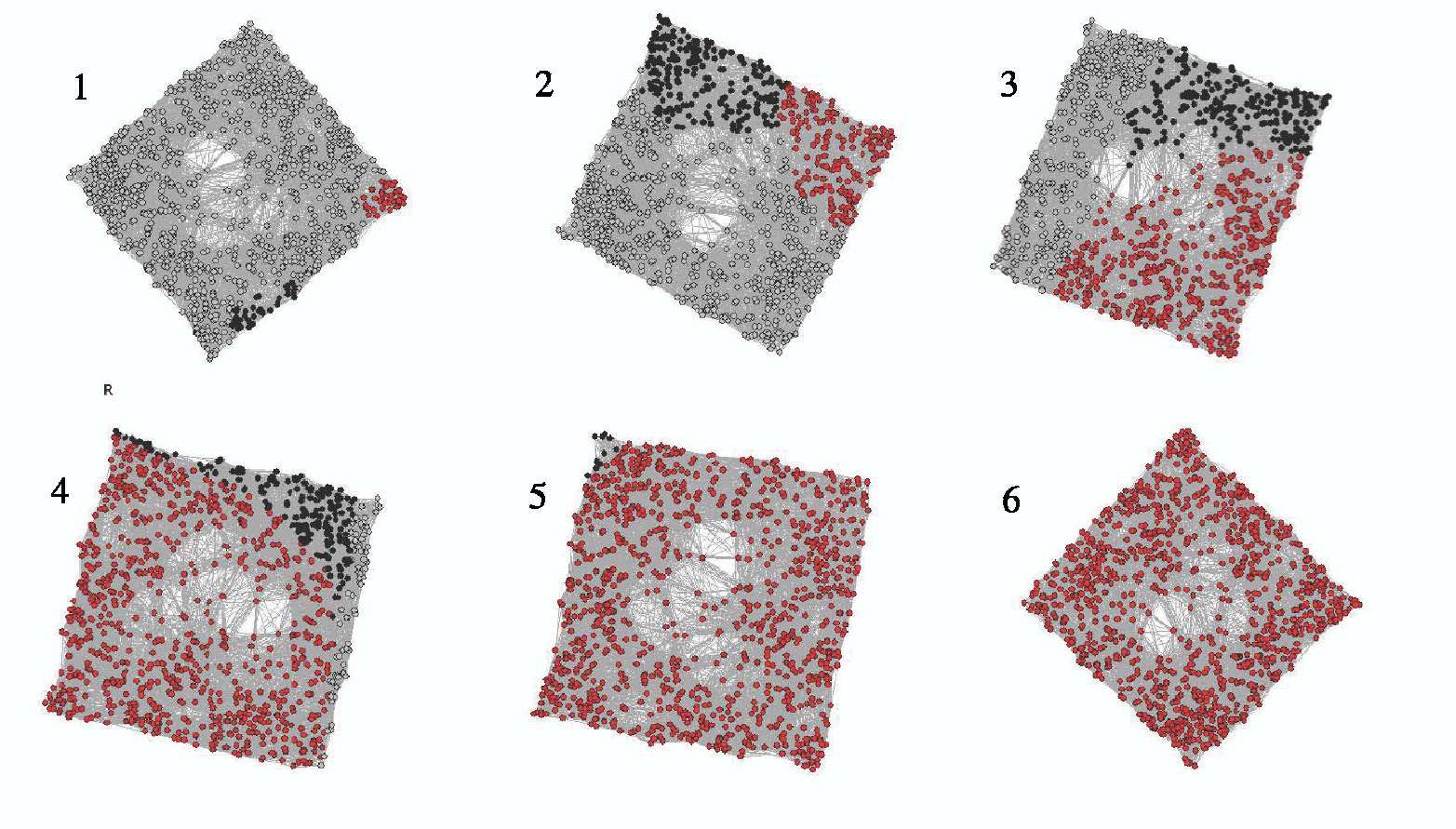
UNIVERSITY OF CAMBRIDGE

# How does suicide compare to voting-based alternatives?

- Much lower storage and communication costs, though signing operation for distributed suicide is computationally expensive
- Decisions are reached more quickly
- No restrictions on node mobility or 'honest majority' assumption
- Requires good nodes to value network's welfare over individual utility
- Enables precision DoS attack: can remove strategic nodes
- Suicide lets an attacker remove one good node for the price of one bad node; in blackballing, a colluding majority can remove good nodes at will

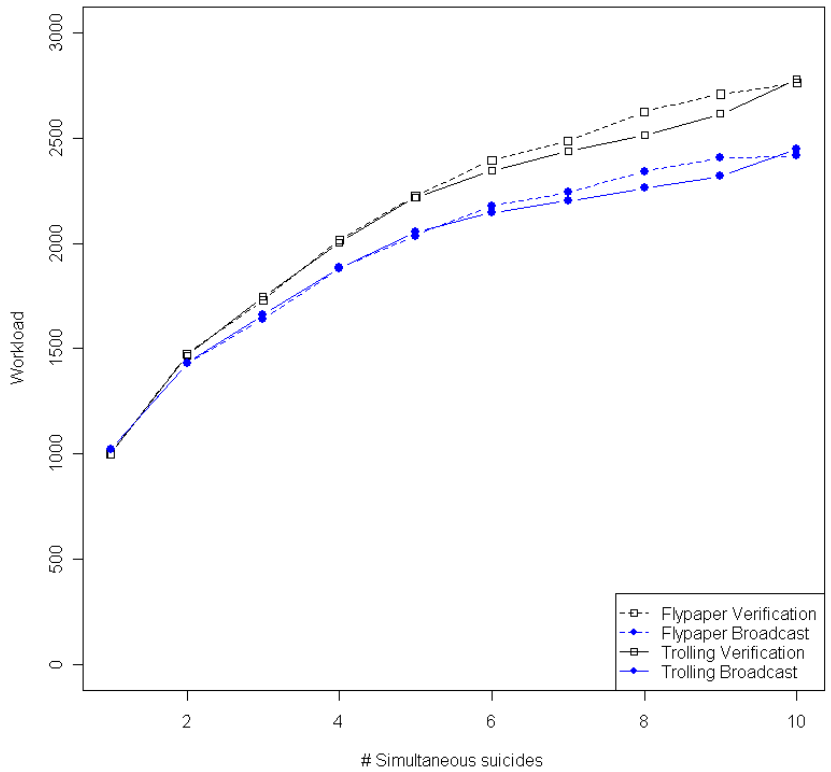
UNIVERSITY OF CAMBRIDGE

# Network performance under multiple suicide offers




UNIVERSITY OF CAMBRIDGE

# Network performance under multiple suicide offers (ctd.)



# Conclusions

- A major challenge for ad-hoc networks is how to remove nodes that are observed to be behaving badly
- Existing threshold voting proposals are computationally expensive, operationally restrictive, and susceptible to manipulation
- We switched from voting against bad nodes to affirming good ones, improving storage costs and enabling a lightweight 'buddy list' protocol
- Suicide is fast, cheap, scalable and handles node mobility
- For more: `http://www.cl.cam.ac.uk/~twm29/`

UNIVERSITY OF CAMBRIDGE