Round-Efficient Broadcast Authentication Protocols for Fixed Topology Classes

Haowen Chan, Adrian Perrig
Carnegie Mellon University

Talk Outline

• Background / Motivation
• Optimizations for the Path Topology
• Summary of Other Results

Talk Outline

• Background / Motivation
• Optimizations for the Path Topology
• Summary of Other Results

Multi-receiver Authentication in Sensor/Ad-hoc Networks

Authentication Methods

• Signature: Sender S signs M using private key
  • Need support for public key crypto
• Multi-receiver Message Authentication Codes
  • Additional O(n) overhead in message size
• TESLA [Perrig et al., 2002]
  • Need time synchronization
• Communication-Efficient with Minimal Assumptions
  • Guy Fawkes [Anderson et al. 1998]
  • Hash Tree-based [Chan & Perrig 2008]

Assumptions

• Sender knows full network topology
• Sender shares a unique symmetric key $K_i$ with each receiver $R_i$
Hash Tree Based Broadcast

- Construct a hash tree with MACs at the leaves
  - \( r \) acts as an authenticator for \( M \)
  - \( L_i = \text{MAC}_{K_i}(M) \)
- Idea: Adversary can’t compute \( r \) for forged \( M' \) since it does not know any of the MAC values of the legitimate nodes

Receiver Verification

- Given Message \( M \), hash tree root vertex \( r \)
- Receiver \( R \), verifies that \( L_i = \text{MAC}_{K_i}(M) \) is a leaf in hash tree with root \( r \)
- Verification path = all siblings on path to root

General Tree Topology: 3 Passes

1. Sender broadcasts message \( M \) with hash tree root \( r \)
2. Receivers reconstruct hash tree with leaves \( L_i = \text{MAC}_{K_i}(M) \)
3. Verification paths disseminated

Path Topology

- Common applications
  - Actual linear topologies (roadway, corridor)
  - Path from leaf to root in spanning tree
  - Along a routing path
  - 1 round = one interaction between neighbors
  - Message from \( S \) to \( R_n \) takes \( n \) rounds
- Unoptimized: 3 passes = \( 3n \) rounds

Talk Outline

- Background / Motivation
- Optimizations for the Path Topology
- Summary of Other Results

Observation

- Can start reconstructing the hash tree immediately upon receiving \( M \)
- “Piggy-back” the two outgoing passes together
  - Achieve \( 2n \) rounds
  - Outgoing pass: left-siblings computed
  - Incoming pass: right-siblings computed
2n-Round Protocol

$S$ precomputes the whole tree

$L_i = \text{MAC}_K(M)$

$S \rightarrow R_1 \rightarrow R_2 \rightarrow R_3 \rightarrow R_4 \rightarrow R_5 \rightarrow R_6 \rightarrow R_7 \rightarrow R_8$

$S \leftarrow M, r$

$R_1, R_2, R_3, R_4, R_5, R_6, R_7, R_8$

$S \leftarrow M, r$

$S \rightarrow R_1 \rightarrow R_2 \rightarrow R_3 \rightarrow R_4 \rightarrow R_5 \rightarrow R_6 \rightarrow R_7 \rightarrow R_8$

$S \leftarrow M, r$

$S \leftarrow M, r$

$S \rightarrow R_1 \rightarrow R_2 \rightarrow R_3 \rightarrow R_4 \rightarrow R_5 \rightarrow R_6 \rightarrow R_7 \rightarrow R_8$

$S \leftarrow M, r$

$S \rightarrow R_1 \rightarrow R_2 \rightarrow R_3 \rightarrow R_4 \rightarrow R_5 \rightarrow R_6 \rightarrow R_7 \rightarrow R_8$
2n-Round Protocol

Further Optimizations

- Computation of Node \( v_6 \) causes delay
- If Sender precomputes and sends \( v_6 \)
  - Nodes 1-4 can build verification paths independently of 5-8
  - Split apart the two subtrees

1.5n-Round Protocol
1.5\(n\)-Round Protocol

\[ S \rightarrow R_1 \rightarrow R_2 \rightarrow R_3 \rightarrow R_4 \rightarrow R_5 \rightarrow R_6 \rightarrow R_7 \]

1.5\(n\)-Round Protocol

\[ S \rightarrow R_1 \rightarrow R_2 \rightarrow R_3 \rightarrow R_4 \rightarrow R_5 \rightarrow R_6 \rightarrow R_7 \]

1.5\(n\)-Round Protocol

\[ S \rightarrow R_1 \rightarrow R_2 \rightarrow R_3 \rightarrow R_4 \rightarrow R_5 \rightarrow R_6 \rightarrow R_7 \]

General Optimization

- Break the receiver set into \(\log n\) groups
- Doubles communication overhead but halves the number of rounds
- No protocol can be faster than this

\(n\)-Round Protocol

\[ \frac{1}{2}n \rightarrow \frac{1}{2}n \quad 1 \frac{1}{2}n \text{ rounds} \]

\[ \frac{1}{2}n \rightarrow \frac{1}{4}n \rightarrow \frac{1}{4}n \quad 1 \frac{1}{4}n \text{ rounds} \]

\[ \frac{1}{2}n \rightarrow \frac{1}{4}n \rightarrow \frac{1}{8}n \rightarrow \frac{1}{8}n \quad 1 \frac{1}{8}n \text{ rounds} \]
Talk Outline

• Background / Motivation
• Optimizations for the Path Topology
• Summary of Other Results

Guy Fawkes on the Path Topology

• Optimization to reduce Guy Fawkes to $2n$ rounds
• Reduce that to $n$ rounds using the same divide-and-conquer technique

Round Complexity Lower Bounds

• Any Signature-free Broadcast Authentication Protocol that completes in $(2-\rho)\log n$ rounds for $0 < \rho \leq 1$ must have $\Omega(n^\rho)$ comm. overhead per node
• Proven using a reduction to a known result for multi-receiver MACs
• Protocols with polylog communication overhead must take $2 \log n$ rounds or more

Tightness of the Bound

• Optimization of protocols for fully connected topologies
• Achieves $2 \log n$ rounds with $O(\log^2 n)$ communication per node
• No protocol with polylog per node communication overhead can take fewer rounds

Lower Bounds for Trees

• Any Signature-free Broadcast Authentication Protocol that completes in $(2.44-\rho)\log n + O(1)$ rounds in a tree topology must have $\Omega(n^\rho)$ comm. overhead per node
• Strictly more than 2 passes are needed for trees
  • Known protocols are likely already optimal for trees

Thank You!

Haowen Chan
haowenchan@cmu.edu