Secure Aggregation for Wireless Networks

Lingxuan Hu, David Evans
Department of Computer Science
University of Virginia

Preview

- Introduction
  - Wireless Sensor Networks (WSN)
  - Data Aggregation
  - Security Risks of Data Aggregation in a WSN
  - SPINS: Security Protocols for Sensor Networks
- Protocol for Secure Aggregation
  - Goals
  - Assumptions
  - Encryption and authentication
  - Data Validation and Integrity

Introduction

- Wireless Sensor Networks (WSN)
  - Consist of sensor devices (nodes)
    - Limited memory, power, and computational abilities
    - Transmit information regarding a hostile environment to a fixed base station
- Data Aggregation
  - Data is gathered and summarized
  - Reduces amount of power consumed when data is forwarded

Introduction

- Security Risks of Data Aggregation in a WSN
  - Intruder nodes may be placed within the network
    - Data can be forged, modified, or discarded
  - A compromised node gives an intruder access to key information
    - Encryption becomes more complicated
      - Suggestion: A key shared between a node and the base station
        - No good because each node needs to be able to understand the messages it receives so that it can perform aggregation
      - Suggestion: A key shared between all the nodes and the base station
        - No good because the intruder will have the key and can control the entire network

Introduction

- SPINS: Security Protocols for Sensor Networks
  - Adrian Perrig, Robert Szewczyk, J.D. Tygar, Victor Wen, David E. Culler
  - Suite of security protocols for sensor networks
  - Consists of two secure building blocks
    - SNEP (Secure Network Encryption Protocol)
      - Data confidentiality
      - Two-party data authentication
      - Evidence of data freshness
    - μTESLA
      - Provides authenticated broadcast for resource-constrained environments
      - Delayed key disclosure
      - Clock synchronization

Protocol for Secure Aggregation

- Goal:
  - Ensure integrity of data transmitted to the base station
    - Efficiently detect node misbehavior
    - A compromised node should not be able to mislead the network about the readings of other nodes
- Not concerned with confidentiality of data

Protocol for Secure Aggregation

- A specific encryption algorithm *is not* required
- An efficient and secure MAC algorithm *is* required
- μTESLA protocol used for authentication
- Assumptions
  - Powerful base station that can broadcast data to all nodes (sensor devices can only communicate with nearby nodes)
  - Low-level network mechanisms exist to provide reliable message delivery
  - Network must be spread out and dense
  - Each node can establish shared secrets with the base station before deployment
  - Additional: A secure self-organizing protocol is used to form a routing hierarchy where each node has an immediate parent (simple hierarchical tree)

Protocol for Secure Aggregation

- Notation

A, B, C, …	sensor nodes
S	base station
$A \rightarrow B$	node A sends a message to B
ID_A	unique ID of node A
$M_1 \mid M_2$	concatenation of messages M_1 and M_2
E(K, M)	encryption of M using key K
MAC(K, M)	authentication code of M using key K

Protocol for Secure Aggregation

- Notation, continued
  - Aggr(x, y) — result of aggregation function on x and y
  - K_{AS} — unique key shared between A and S
  - R_A — data reading value of node A
  - K_{Ai} — the ith key for node A = $E(K_{AS}, i)$
  - K_i — the ith key in the key chain = $F^{n-i}(K)$
  - F — public one way function
  - i — time interval
  - n — number of applications of F to a secret K

Protocol for Secure Aggregation

- Exploits two main ideas
  - Delayed aggregation
    - Messages are forwarded unchanged over the first hop and are aggregated at the second hop
      - Ensures that a compromised node cannot tamper with many sensor readings
  - Delayed authentication
    - Messages are authenticated after a time delay
      - Saves resources
      - Allows authentication keys to be symmetric keys that are revealed after the time delay has expired

Protocol for Secure Aggregation

- Encryption and authentication
  - Base Station → Sensor Devices
    - Base station generates a one-way key chain using F
    - Before deployment, each sensor stores the key for the first time interval, K_0

- $K_i = F(K_{i+1})$
  - $K_0 = F(K_1)$ $= F(F^{n-1}(K)) =$
  - $K_1 = F(K_2)$ $= F(F^{n-2}(K)) =$
  - $K_2 = F(K_3)$ $= F(F^{n-3}(K)) =$
  - $K_3 = F(K_4)$ $= F(F^{n-4}(K)) =$

- $K_i = F^{n-i}(K)$
  - $K_0 = F^n(K)$
  - $K_1 = F^{n-1}(K)$
  - $K_2 = F^{n-2}(K)$
  - $K_3 = F^{n-3}(K)$

Protocol for Secure Aggregation

- Encryption and Authentication
  - Base Station → Sensor Devices
    - The first transmissions from the base station are encrypted with K_1
    - After some time has passed and all messages have been transmitted and received, each sensor calculates:
      - $F(K_1) = F(F^{n-1}(K)) = F^n(K)$
    - and verifies:
      - $K_0 = F^n(K) = F(K_1)$
    - *If $K_0 = F(K_1)$, then the sensor can decrypt messages that were transmitted earlier and encrypted with K_0*
    - All remaining keys, up to $K_n = K$, are revealed the same way

Protocol for Secure Aggregation

- Encryption and Authentication
  - Sensor Device → Base Station, Sensor Device
    - Before deployment, each sensor stores a symmetric secret key, K_{AS}
    - Temporary node encryption keys are computed by encrypting counter values with K_{AS}
      - Ex. $K_{A0} = E(K_{AS}, 0)$
    - Base station synchronizes counter values with devices
    - Upon receipt of messages from sensors, the base station broadcasts the temporary node encryption keys
    - A sensor retrieves the keys it needs and authenticates the messages received from nearby sensors
    - Sensors then advance to the next temporary node encryption key

Protocol for Secure Aggregation

- Data Validation and Integrity
  - Base station authenticates messages
    - compute MACs using temporary node encryption keys
    - Compare computed MACs to transmitted MACs
    - Verify that they match
  - Each node is responsible for data validation
    - Base station broadcasts temporary node encryption keys
    - Nodes listen for and retrieve necessary keys
    - Nodes authenticate and validate messages received from nearby nodes
      - A forged message is detected if a parent's calculated MAC is inconsistent with a MAC received from a child or grandchild
      - Parent nodes send alarm messages when a forged message is detected

Protocol for Secure Aggregation

$A \rightarrow E$ $ID_A \mid R_A \mid MAC(K_{Ai},R_A)$

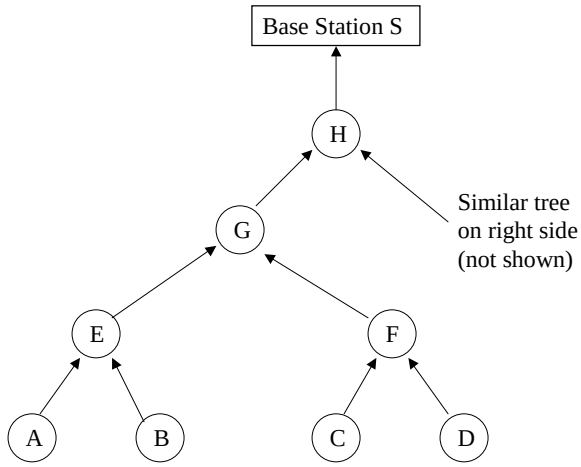
$B \rightarrow E$ $ID_B \mid R_B \mid MAC(K_{Bi},R_B)$

$C \rightarrow F$ $ID_C \mid R_C \mid MAC(K_{Ci},R_C)$

$D \rightarrow F$ $ID_D \mid R_D \mid MAC(K_{Di},R_D)$

$E \rightarrow G$ $ID_A \mid R_A \mid MAC(K_{Ai},R_A)$
$\mid ID_B \mid R_B \mid MAC(K_{Bi},R_B)$
$\mid MAC(K_{Ei}, Aggr(R_A, R_B))$

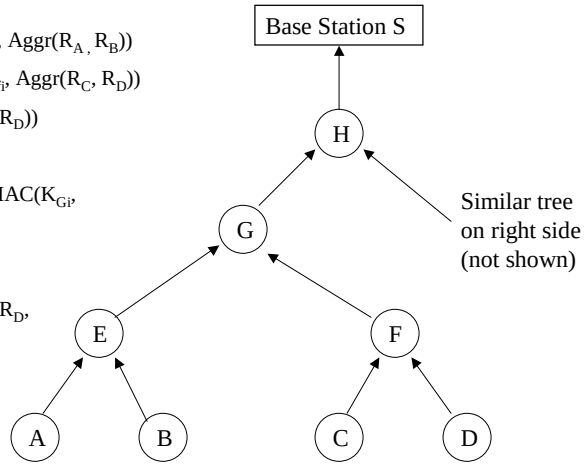
$F \rightarrow G$ $ID_C \mid R_C \mid MAC(K_{Ci},R_C)$
$\mid ID_D \mid R_D \mid MAC(K_{Di},R_D)$
$\mid MAC(K_{Fi}, Aggr(R_C, R_D))$



Protocol for Secure Aggregation

$G \rightarrow H$ $ID_E \mid Aggr(R_A, R_B) \mid MAC(K_{Ei}, Aggr(R_A, R_B))$
$\mid ID_F \mid Aggr(R_C, R_D) \mid MAC(K_{fi}, Aggr(R_C, R_D))$
$\mid MAC(K_{Gi}, Aggr(R_A, R_B, R_C, R_D))$

$H \rightarrow S$ $ID_G \mid Aggr(R_A, R_B, R_C, R_D) \mid MAC(K_{Gi},$
$Aggr(R_A, R_B, R_C, R_D))$
$\mid$ … same from right side
$\mid MAC(K_{Hi}, Aggr(R_A, R_B, R_C, R_D,$
… readings from right side))



Review

- Introduction
  - Wireless Sensor Networks (WSN)
  - Data Aggregation
  - Security Risks of Data Aggregation in a WSN
  - SPINS: Security Protocols for Sensor Networks
- Protocol for Secure Aggregation
  - Goals
  - Assumptions
  - Encryption and authentication
  - Data Validation and Integrity
- What's Next:
  - Attack Analysis
  - Scalable Variation