A Trust Based Framework for Secure Data Aggregation in Wireless Sensor Networks

Wei Zhang, Sajal K. Das, and Yonghe Liu
Center for Research in Wireless Mobility and Networking (CReWMaN)
Department of Computer Science and Engineering, The University of Texas at Arlington
This paper comes from IEEE SECON 2006 proceedings

Outline

I. Motivation
II. Preliminaries
A Trust Based Framework Against False Data Injection
IV. Experiment

I. Motivation

This paper proposes an integrative, systematic approach to secure aggregation process against compromised node attacks and quantify uncertainty in the aggregation results. Instead of solely relying on cryptographic techniques.

Motivation (cont)

This paper proposes a trust based framework. The trustworthiness (reputation) of each individual sensor node is evaluated by using an information theoretic concept, Kullback-Leibler (KL) distance, to identify the compromised nodes. Upon aggregating, an opinion, a metric of the degree of belief, is generated to represent the uncertainty in the aggregation result.

Motivation (cont)

As the result is being disseminated and assembled through the routes to the sink, this opinion will be propagated and regulated by Josang’s belief model. Following this model, the uncertainty within the data and aggregation results can be effectively quantified throughout the network.

II. Preliminaries

1. System Model
2. Threat Model
3. Josang’s Belief Model

1. System Model

A large number of densely deployed sensors which are organized into clusters using some clustering scheme.
1. System Model (cont)

A cluster head acts as a gateway of the cluster and forwards aggregated results to the sink (BS).

- Each sensor has bidirectional communication capability and can directly communicate with its cluster head.
- In one cluster, all sensor nodes including the cluster head and aggregators are physically proximate and hence their sensory data is highly correlated.

2. Threat Model

An adversary can compromise any sensor nodes.

- Once a sensor node is compromised, all secret information is disclosed. As a result, the adversary can manipulate the compromised nodes and alter/forge sensory data to disrupt normal network operations.

3. Josang’s Belief Model

The definition of opinion is given as follows.

Definition: An opinion $\omega = (b, d, u, a)$, is a quadruple where the components respectively correspond to belief, disbelief, uncertainty, and relative atomicity, such that $a, b, d, u \in [0, 1]$ and $b + d + u = 1$

The relative atomicity ($a$) is used for computing an opinion’s probability expectation as $O = E(\omega) = b + au$.

III. A Trust Based Framework Against False Data Injection

1. Overview of the Framework
2. Aggregator
3. Cluster Head
4. Sensor Nodes (Cluster Members)
5. Reselection
1. Overview of the Framework

- Clusters are formed through existing clustering algorithm such as LEACH.
- The cluster head randomly selects a set of nodes as aggregators, and partitions the cluster into multiple separate aggregating sets.
- The number of aggregators depend on the cluster’s density and desirable data accuracy.
- The cluster head broadcasts aggregators’ information to all sensor nodes within the cluster.

At the same time, all sensor nodes can overhear the reports sent by the aggregators and cluster head so they can evaluate and maintain reputations of the aggregators and cluster head according to their own judgment.

2. Aggregator

- Reputation computation and updating
- Reputation classification and aggregation
- Opinion formulation and forwarding to cluster head

How to exclude outliers

- In order to exclude outliers without the knowledge of the true value, an aggregator shall compare each data point with the median of these samples as the median is more robust. A sample value significantly deviated from the median will be deemed as an outlier and expelled from aggregation.

According to central limit theorem, these sensing data will approximately follow normal distribution: \( N(\mu, \sigma) \), where \( \mu \) is the mean and \( \sigma \) is the standard deviation.

- Sampling data,
- Excluding outliers,
- Maintaining nodes’ reputations,
- Calculating aggregation result,
- Forming its opinion about the result,
- Reporting the result along with opinion to cluster head.
For normal distribution, empirical theory shows that about 68% of the samples fall within one standard deviation of the mean, i.e., between \( \mu - \sigma \) and \( \mu + \sigma \).

When there is no compromised node and all nodes have similar capability, each node's data has the same probability (0.68) of falling within the above range for a particular sampling round.

We term this ideal node frequency distribution which actually is Bernoulli distribution with probability of 0.68. We term the actual frequency of a node's data falling into the above range actual node frequency.

**NOTICE**

computing node's reputation

- The difference between ideal node frequency distribution and actual node frequency distribution can be measured by a distance.
- This distance can be employed as a measure to represent a node's reputation.

we employ Kullback-Leibler (KL) distance, or relative entropy to quantify the distance.

\[
D(p||q) = (1 - p) \log \frac{1 - p}{1 - q} + p \log \frac{p}{q}
\]

\( p, q \in [0,1] \)

reputation is defined as:

\[
r = \frac{1 - D(p||q)}{1 + D(p||q)}
\]

\( r \in [0,1] \)

an example

- Two sensors s1 and s2 in one aggregating set.
- For s1, its actual frequency after a particular sampling round, say \( t_1 \), is \( f_{s1}^{t_1} = 0.61 \).
- According to Equation (1),
  
  \[
  D(f_{s1}^{t_1}||f_{ideal}^{t_1}) = 0.0023
  \]
  
  Then, its reputation is \( r(s_1^{t_1}) = 0.940 \).

- While for s2, \( f_{s2}^{t_1} = 0.63 \).
  
  Then, \( D(f_{s2}^{t_1}||f_{ideal}^{t_1}) = 0.0081 \)
  
  \( r(s_2^{t_1}) = 0.919 \).

Updating node's reputation with time

- After several rounds, say \( t_2 \), \( f_{s1}^{t_2} = 0.68 \) and \( f_{s2}^{t_2} = 0.30 \).
  
  At this moment, \( D(f_{s1}^{t_2}||f_{ideal}^{t_2}) = 0 \) and \( r(s_1^{t_2}) = 1.0 \).
  
  While, \( D(f_{s2}^{t_2}||f_{ideal}^{t_2}) = 0.436 \), and \( r(s_2^{t_2}) = 0.902 \).
Reputation classification and aggregation

1. Reputation classification
2. Aggregation

Reputation classification

- The K-Means partition algorithm is employed for reputation classification.
- The basic idea of the K-Means algorithm is to partition a data set into K disjoint groups to minimize the sum-of-squares criterion.

For example

- In the above example, at time t1, s1 and s2 (reputations are 0.95 and 0.92, respectively) are in the same group.
- At time t2, s1 and s2 (reputations are 1.0 and 0.6, respectively) will be classified into two groups.

Opinion formulation and forwarding to cluster head

1. Opinion formulation
2. Opinion forwarding to cluster head

Opinion formulation

- For an aggregation result X, aggregator A forms its opinion:
  \[ \omega_A^X = \{a^1, a^2, \ldots, a^n\} \]
The expectation of an aggregator’s opinion about the aggregation result $X$

$$\omega_1 = \mathbb{E}(\omega_1) = b^* + \sigma^* u^*/\sqrt{u^*}$$

For example

Suppose that there are 32 sensor nodes in aggregator $A_1$’s aggregating set.

For a particular aggregation result $X_1$, there are 22 sensor nodes whose data fall within the range while the average reputation of the rest is 0.90.

So, $A_1$’s opinion about the result is:

$$\omega_{A_1} = 0.696, 0.132, 0.90$$

This quantifies the uncertainty in the aggregation result from aggregator $A_1$’s viewpoint.

Opinion forwarding to cluster head

- Upon obtaining the aggregation result and opinion, the aggregator will send them together as its report to cluster head.
- Besides, the aggregator periodically broadcasts its reputation list so that all the sensor nodes and the cluster head are aware of others nodes’ reputations.

3. Cluster Head

- **Opinion formulation**
- **Belief discounting**
- **Final result combination and belief consensus**

The cluster head maintains its opinion toward each aggregator.

The opinion here can be also considered as an aggregator’s reputation from the cluster head’s viewpoint.

- The cluster head evaluates an aggregator’s reputation by examining consistency between its own sensing data and the results from all aggregators.
- The cluster head trusts the value that most aggregators agree on and considers them as “honest” while treats the rest as “dishonest”.
- The cluster head can judge each aggregator’s behavior upon receiving reported result.

The posterior probability estimates of these binary events (being honest or dishonest) can be represented by the Beta distribution with two parameters $\alpha$ and $\beta$, using the gamma function $\Gamma$ as [8]:

$$\omega_{A_1} = \text{Beta}(0.696, 0.132, 0.90)$$
The probability expectation value of the Beta distribution is given by:

\[ E(p) = \frac{\alpha}{\alpha + \beta} \]

The cluster head’s opinion about aggregator A is given as:

\[ \omega_A^N = (\theta_A^N, \phi_A^N, \psi_A^N, \nu_A^N) \]

Let \( h_A^N \) be the number of events that an aggregator’s result is honest observed by the cluster head, and \( i_A^N \) be the contrary.

\[ h_A^N = \frac{\nu_A^N}{\nu_A^N + \phi_A^N + \psi_A^N} \]

\[ d_A^N = \frac{\psi_A^N}{\nu_A^N + \phi_A^N + \psi_A^N} \]

\[ w_A^N = \frac{\phi_A^N}{\nu_A^N + \phi_A^N + \psi_A^N} \]

\[ e_A^N = 0.5 \]

For example

Assume upon the 100th sampling round, a cluster head observes 97 “honest” events from aggregator A1, thus

\[ \omega_A^N = (0.951, 0.029, 0.02, 0.5) \]

and \( \omega_A^N = 0.827 \).

Similarly, aggregator A2 has 60 “honest” events, then

\[ \omega_A^N = (0.59, 0.39, 0.02, 0.5), \]

and \( \omega_A^N = 0.6 \).

Belief discounting

When the cluster head receives an aggregator’s report, the cluster head discounts it according to its opinion toward this aggregator.

\[ \omega_A^N = (\theta_A^N, \phi_A^N, \psi_A^N, \nu_A^N) \]

Then

\[ \omega_A^N = (\theta_A^N, \phi_A^N, \psi_A^N, \nu_A^N) \]
The expectation of the cluster head’s opinion about the aggregator’s report is
\[ \omega H A = b H A + w H \times u H A. \]

For example
- At above the 100th sampling round, the cluster head’s opinion about aggregator A1 is
  \[ \omega_{H1}^H = (0.061, 0.029, 0.02, 0.9) \]
  And A1’s opinion about the result is
  \[ \omega_{A1}^A = (0.688, 0, 0.312, 0.9) \]

- The final aggregation result is calculated by
  \[ X = w_1 X_1 + w_2 X_2 \]
  \[ w_1 \text{ and } w_2 \text{ are weighing factors.} \]

- Similarly,
  \[ \omega_{H1} = \frac{\omega_{H1}^H \cdot \omega_{A1}^A}{\omega_{H1}^H \cdot \omega_{A1}^A + \omega_{H1}^H \cdot \omega_{A1}^A} \]

- For the final result combination and belief consensus
  The final aggregation result is calculated by
  \[ X = u H A_1 + u H A_2 - u H A_1 \times u H A_2 \]
  \[ \omega_{H1} = \frac{\omega_{H1}^H \cdot \omega_{A1}^A}{\omega_{H1}^H \cdot \omega_{A1}^A + \omega_{H1}^H \cdot \omega_{A1}^A} \]

- The final aggregation result is calculated by
  \[ X = w_1 X_1 + w_2 X_2 \]
  \[ w_1 \text{ and } w_2 \text{ are weighing factors.} \]

- Similarly,
  \[ \omega_{H1} = \frac{\omega_{H1}^H \cdot \omega_{A1}^A}{\omega_{H1}^H \cdot \omega_{A1}^A + \omega_{H1}^H \cdot \omega_{A1}^A} \]

- For the final result combination and belief consensus
  The final aggregation result is calculated by
  \[ X = u H A_1 + u H A_2 - u H A_1 \times u H A_2 \]
  \[ \omega_{H1} = \frac{\omega_{H1}^H \cdot \omega_{A1}^A}{\omega_{H1}^H \cdot \omega_{A1}^A + \omega_{H1}^H \cdot \omega_{A1}^A} \]

- For the final result combination and belief consensus
  The final aggregation result is calculated by
  \[ X = u H A_1 + u H A_2 - u H A_1 \times u H A_2 \]
  \[ \omega_{H1} = \frac{\omega_{H1}^H \cdot \omega_{A1}^A}{\omega_{H1}^H \cdot \omega_{A1}^A + \omega_{H1}^H \cdot \omega_{A1}^A} \]

- For the final result combination and belief consensus
  The final aggregation result is calculated by
  \[ X = u H A_1 + u H A_2 - u H A_1 \times u H A_2 \]
  \[ \omega_{H1} = \frac{\omega_{H1}^H \cdot \omega_{A1}^A}{\omega_{H1}^H \cdot \omega_{A1}^A + \omega_{H1}^H \cdot \omega_{A1}^A} \]
For example
\[
\begin{align*}
\bar{w}_1^{B_A} &= (0.674, 0.364, 0.002) \\
\bar{w}_2^{B_A} &= (0.364, 0.674, 0.002) \\
\bar{w}_1^{B_B \cup B_C} &= (0.074, 0, 0.926) \\
\bar{w}_2^{B_B \cup B_C} &= 0.926
\end{align*}
\]

4. Sensor Nodes (Cluster Members)
- The sensor nodes overhear the report sent by an aggregator or cluster head and update corresponding aggregator's or cluster head's reputation by counting the numbers of "honest" events, according to their own readings.

IV. Reselection
- To balance energy consumption
- To prevent from a compromised aggregator or cluster head

V. Experiment
Four factors are considered for false data injection attacks:
1) data value sent by compromised nodes;
2) time duration for sending false data by compromised nodes;
3) total number of compromised nodes in one aggregating set;
4) misbehavior pattern of the compromised nodes.

<table>
<thead>
<tr>
<th>Test Case No.</th>
<th>Malicious Node Domain (%)</th>
<th>Malicious Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case 1</td>
<td>100</td>
<td>2</td>
</tr>
<tr>
<td>Case 2</td>
<td>200</td>
<td>2</td>
</tr>
<tr>
<td>Case 3</td>
<td>50</td>
<td>1</td>
</tr>
<tr>
<td>Case 4</td>
<td>10</td>
<td>1</td>
</tr>
</tbody>
</table>

Fig. 4. Reputation of sensor nodes

Fig. 5. Evolution of opinion

Fig. 6. Aggregate results for Case 3

Fig. 7. Aggregate results for Case 1 and Case 2

Fig. 8. Aggregate results for Case 3 and Case 4

Fig. 9. Aggregate results for Case 1, 2, 3, 4