SMART: A SVM-based Misbehavior Detection and Trust Management Framework for MANETs

Wenjia Li
Anupam Joshi
Tim Finin

May 18th, 2010
Outline

• Introduction and Motivation
• Related Work
• Outlier Detection Algorithm
• Performance Evaluation
• Conclusion and Future Work
Introduction

• Basic features of mobile ad hoc networks
  • Open and unreliable transmission medium
    • Data are *easily* disclosed to unwanted third parties
  • Node mobility and constantly changing topology
    • Data communication may be *frequently* disrupted
  • Absence of pre-deployed infrastructure
    • *Selfish* nodes refuse to forward packets for others → System performance severely downgraded
  • Limited power supply for each node
    • Consequence: *SHORT* transmission range and *LIMITED* computation capability

⇒ *Ad Hoc Networks are extremely vulnerable* to various misbehaviors
Misbehavior Detection

• An important method to protect MANETs from BOTH external attackers AND internal compromised nodes

• Current misbehavior detection mainly relies on a predefined threshold to detect misbehaviors
  • Threshold-based intrusion detection: set a threshold for each kind of misbehavior, and alert only when the number of misbehaviors exceeds the threshold

• Drawbacks of threshold
  • Hard to define beforehand
    • How can we accurately predict an adversary’s attack pattern?
  • Need to adjust frequently
    • Adjust according to the changing topology and node mobility
  • Easy to exploit
    • A smart adversary reduces its misbehaviors JUST below the threshold
Motivating Scenario I

(a) Observer 1
- A: Well-behaved
- B: Misbehaving
- # of Packets: Dropped: 3, # of Incorrect Opinions: 4

(b) Observer 2
- A: Well-behaved
- C: Misbehaving
- # of Packets: Dropped: 4, # of RTS flooding packets: 2

- # of Packets: Dropped: 9, # of RTS flooding packets: 6
Trust Management

- Evaluation of how *trustworthy* a node will be based on the *observations* to its previous behaviors

- Two types of observations
  - *Direct* observation (First-hand information)
    - Observations made by a node ITSELF
  - *Indirect* observation (Second-hand information)
    - Observations obtained from OTHER nodes

- Majority of current trust management schemes model trust in form of *ONE single scalar*
  - Observations to all types of misbehaviors are used to derive ONE single trust value for each node
  - Neither expressive nor accurate in complicated scenarios
Motivating Scenario II

Ten incoming packets $A_i$ to Ten outgoing packets $B_j$.

1: Ten RTS flooding frames

2: Ten packets modified

3: Ten incorrect observations for others

(a)

Trust$_1$ = Trust$_2$ = Trust$_3$

(b)
SMART

- SVM-based misbehavior detection and trust management framework
  - *Outlier detection* technique is used for behavioral data collection
    - Misbehaviors generally *deviate* from normal behaviors → misbehaving nodes can be viewed as *outliers*
  - *SVM* technique is used to identify misbehaving nodes
    - Incorporate previous knowledge on misbehaviors to detect *unknown* misbehaviors
- *Multi-dimensional* trust management
  - Divide trustworthiness into several dimensions (for example 3)
  - Each dimension of trustworthiness is derived by a subset of misbehaviors
Related Work

• Misbehavior detection in MANETs
  • *Selfish* node VS. *malicious* node
    • Selfishness: Merely want to preserve resource (battery power, bandwidth), so may deny forwarding packets or route requests
    • Maliciousness: Intentionally want to disrupt the network services, so may take any action to meet this goal
      • Packet drop, modification, misroute, fake RTS requests, etc.
  • Intrusion detection system (IDS) for MANETs
    • IDS sensor deployed on each node
      • **NOT** energy-efficient
    • Cluster-based IDS by Huang et al.
  • Efforts to identify routing misbehaviors
    • “Watchdog” & “Pathrater” by Marti et al.
Related Work (Cont.)

- Trust management for MANETs
  - Goal: to evaluate behaviors of other nodes and consequently decide the *trustworthiness* for each node based on the behavior assessment

- Various trust management schemes
  - CONFIDANT by Buchegger et al.
    - Four components: a Monitor, a Reputation System, a Trust Manager, and a Path Manager
  - CORE by Michiardi et al.
    - Identifies *selfish* nodes and forces them to cooperate
    - Only *positive* observations are exchanged
  - Node evaluation scheme by Ren et al.
    - Second-hand observations only shared by a subset of neighbors ("trustworthy neighbors")
Framework Overview

- Four components
  - Behavioral data collection
  - Trust management
  - SVM-based misbehavior detection
  - Misbehavior mitigation

- Two stages
  - Training stage
  - Detection stage
Behavioral Data Collection

- Five steps for the **training** stage
  - Neighbor observation
  - Local view formation
  - Local observation exchange
  - View combination
  - Global view formation and data labeling

- One step for **detection** stage
  - Neighbor observation
Local observation and view formation

Incoming Packet

Outgoing Packet B

Abuse RTS requests

Notes:
- PDR – Packet Drop Rate
- PMR – Packet Modification Rate
- RTS – RTS Rate

PDR PMR RTS
1 0.30 0 0
2 0 0.26 0
3 0 0 0.17

Radio Range

Observer

Observed Nodes

1: Packet Dropped
2: Packet Modified
3: RTS Flood
View Combination

• Motivation: **Node mobility** and **channel collision** can make the neighbor observation results **inaccurate**
  • Example scenario

 Observer

 Observed Nodes

 Incoming Packet A

 1

 Radio Range

 1: Packet Dropped

 Outgoing Packet A

 Observer

 Observed Nodes
Dempster-Shafer Theory (DST): combine separate pieces of observations (evidences) to calculate the probability of malicious behaviors

- **Basics:** lack of evidence can **NOT** be viewed as the *refutation* to this evidence
- **Solution:** a node can either hold a *positive* opinion or have *no* opinion to the misbehavior of its neighbor
- **NO** opinion is called “Environment” in this case ($\Theta$)
- How to combine separate pieces of evidences – **Dempster’s Rule of combination**

\[
m_B(A) \oplus m_C(A) = \frac{\sum_{q,r: \alpha_q \cap \alpha_r = A} m_B(\alpha_q)m_C(\alpha_r)}{1 - \sum_{q,r: \alpha_q \cap \alpha_r = \Phi} m_B(\alpha_q)m_C(\alpha_r)}
\]
Multi-dimensional Trust Management

- All trust values initialized to be 1 (trust value $\in [0, 1]$)
- Trust values adjusted based on observation results
  - Both first-hand information and second-hand information
- Three dimensions of trust
  - Collaboration trust
  - Behavioral trust
  - Reference trust
SVM-based Misbehavior Detection

- Three steps in detection stage
  - Behavioral data combination
  - Misbehavior detection
  - Behavioral data update and exchange
Performance Evaluation

- Simulation setup

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation area</td>
<td>600m ( \times ) 600m</td>
</tr>
<tr>
<td>Number of nodes</td>
<td>50, 100, 200</td>
</tr>
<tr>
<td>Transmission range</td>
<td>60m, 90m, 120m</td>
</tr>
<tr>
<td>Simulation Duration</td>
<td>900s</td>
</tr>
<tr>
<td>Mobility pattern of nodes</td>
<td>Random waypoint</td>
</tr>
<tr>
<td>Number of bad nodes</td>
<td>5, 10, 20</td>
</tr>
<tr>
<td>Node motion speed</td>
<td>5m/s, 10m/s, 20m/s</td>
</tr>
</tbody>
</table>
Performance Metrics

- Four metrics
  - Precision
  - Recall
  - Communication Overhead (CO)
  - Convergence Time (CT)

\[ TNPOD = \text{Total Number of Packets for Outlier Detection} \]

\[ CO = \frac{TNPOD}{\text{Total Number of Packets in the network}} \]

\[ CT = \text{Time taken to form a consistent global view of outliers} \]
Simulation Scenario I

Precision and Recall with Different Number of Nodes
(Area: 600m $\times$ 600m, Radio Range: 120m, Speed: 5m/s)
Simulation Scenario II

Precision and Recall with Different Motion Speed
(Area: 600m × 600m, Radio Range: 120m, Num of Nodes: 100)
Simulation Scenario III

Precision and Recall with Different Radio Ranges
(Area: 600m × 600m, Num of Nodes: 100, Node Speed: 5m/s)
Simulation Scenario IV

CR with Different Percentage of Bad Nodes
(Num. of Nodes: 100, Area: 600m × 600m, Range: 120m, Speed: 5m/s)
Adversary Model

- Three adversary models
  - Short-term
  - Ever-changing
  - Comprehensive

<table>
<thead>
<tr>
<th>Node ID</th>
<th>Start</th>
<th>End</th>
<th>Drop</th>
<th>Modify</th>
<th>RTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>10s</td>
<td>180s</td>
<td>90%</td>
<td>10%</td>
<td>0</td>
</tr>
<tr>
<td>33</td>
<td>100s</td>
<td>400s</td>
<td>0</td>
<td>30%</td>
<td>70%</td>
</tr>
<tr>
<td>34</td>
<td>50s</td>
<td>200s</td>
<td>40%</td>
<td>30%</td>
<td>30%</td>
</tr>
<tr>
<td>44</td>
<td>400s</td>
<td>660s</td>
<td>0%</td>
<td>50%</td>
<td>50%</td>
</tr>
<tr>
<td>45</td>
<td>350s</td>
<td>600s</td>
<td>20%</td>
<td>0</td>
<td>80%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Node ID</th>
<th>Start</th>
<th>End</th>
<th>Drop</th>
<th>Modify</th>
<th>RTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>0s</td>
<td>200s</td>
<td>10%</td>
<td>70%</td>
<td>20%</td>
</tr>
<tr>
<td>16</td>
<td>200s</td>
<td>400s</td>
<td>50%</td>
<td>0</td>
<td>50%</td>
</tr>
<tr>
<td>16</td>
<td>400s</td>
<td>900s</td>
<td>90%</td>
<td>10%</td>
<td>0</td>
</tr>
<tr>
<td>33</td>
<td>0s</td>
<td>400s</td>
<td>0</td>
<td>40%</td>
<td>60%</td>
</tr>
<tr>
<td>33</td>
<td>400s</td>
<td>900s</td>
<td>30%</td>
<td>30%</td>
<td>40%</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Node ID</th>
<th>Start</th>
<th>End</th>
<th>Drop</th>
<th>Modify</th>
<th>RTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0s</td>
<td>900s</td>
<td>80%</td>
<td>20%</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0s</td>
<td>900s</td>
<td>0</td>
<td>50%</td>
<td>50%</td>
</tr>
<tr>
<td>3</td>
<td>0s</td>
<td>900s</td>
<td>30%</td>
<td>30%</td>
<td>40%</td>
</tr>
<tr>
<td>4</td>
<td>0s</td>
<td>900s</td>
<td>20%</td>
<td>10%</td>
<td>70%</td>
</tr>
<tr>
<td>5</td>
<td>0s</td>
<td>900s</td>
<td>10%</td>
<td>0</td>
<td>90%</td>
</tr>
</tbody>
</table>
Simulation Scenario V

Effect of Different Adversary Models
Conclusion and Future Work

- SVM-based misbehavior detection and trust management framework for MANETs
  - Outlier detection for behavioral data collection
  - SVM for misbehavior detection
  - Multi-dimensional trust management

- Several simulation scenarios have validated the correctness and efficiency of our approach

- Future work
  - How to properly determine trustworthiness DIRECTLY from the SVM classifier
  - Incorporation of context information to SVM classifier
The End

- Questions
- Comments😊