Trust and Reputation in Wireless Networks

Pull Down Their Genes:
Hong Jai Cho
Jon Francis
Lincoln Roop
Erik Schmidt
Introduction

• Collaboration of wireless devices introduces security threats.
• Soft security threats from malicious users
• Hard security, designed to protect data and system integrity might not be enough
• Soft security threats are most effectively countered with TRM
What are Trust and Reputation?

• Trust: Subjective expectation of a node receiving positive outcomes from another node in a specific context.

• Reputation: Global perception of a node's trustworthiness in a network
Individual-Level Trust Models

Evidence Space
- First-hand Evidence
- Second-hand Evidence

Aggregated Evidence

Trust Space
- Trust and Reputation Evaluation

Interaction Decision making

Interaction Outcome Evaluation
Individual-Level Trust Models: Bootstrapping

- Initial phase of trust & reputation formation
- Accumulate information by interacting with other node
- Dummy requests could be made to speed up building trust evidence
Individual-Level Trust Models: Evidence Space

- Trust model based on past experience form evidence space
- Cumulative weights of positive and negative outcomes form evidence
- $R_{\text{updated}} = p_1 r_{\text{past}} + p_2 r_{\text{latest}}$
- Some TRMs use $p_1 >> p_2$, some $p_2 >> p_1$
- Some make reputation "hard to earn but easy to lose."
Individual-Level Trust Models: Aggregation of Second Hand Evidences

- Lack of direct observation of some nodes compensated by second hand trust information
- False second hand evidence can skew evaluation:
  - badmouthing
  - ballot stuffing
- Credibility of witness nodes are used to evaluate second hand trust information
- Deviation test performed on each second hand info
  - Assumption: pattern of subject node is more likely to remain same
Individual-Level Trust Models: Mapping from Evidence Space to Trust Space

- Beta distribution is commonly used to model evidence space and to map it to trust space.
- Simple mapping can introduce errors because of uncertainties in highly dynamic wireless networks:
  - Certainty increases as evidences increase.
  - Certainty decreases as extent of conflicts increase.
- People have prosed ways to address both increase of evidences and conflicts (refer to Yu et al. for more info).
Individual-Level Trust Models: Interaction Decision Making

- Three common decision making methods
  - Threshold-Based Method:
    - If the trustworthiness of a node is below a threshold, it is not considered for interaction.
  - Ranking-Based Method:
    - Trustworthiness values can be used to rank nodes. Cannot apply to service composition.
  - Weight-Based Method:
    - \[ D = \sum (r_i \times d_i) \]
Individual-Level Trust Models: Interaction Outcome Evaluation

- Most common approach of evaluation is to use dedicated watchdog nodes or mobile agents.
- Watchdogs can be dedicated node or regular nodes that evaluate interactions of other nodes.
- Mobile agents migrate through networks, form overlay on top of network, perform the task of watchdogs.
- In WSN decision/evaluation is often estimation based on collective sensor reports.
System-Level Trust Models

• Effective system level model should have evidence dissemination module
  ○ For quick sharing of past observations

• Effective system level model should have punitive measures to deter maliciousness.
  ○ e.g. isolate selfish node

• In some WSN like MANETs, it is rational for nodes to act selfishly (power restraint)
System-Level Trust Models

System-level Trust Model

- Trust-based Reward/Punishment Module
- Trust Evidence Dissemination Module

Trust/reputation values

Individual-level Trust Model
System-Level Trust Models: Games

- Cooperation stimulation mechanisms needed to keep the naturally non-cooperative nodes to cooperate
  - Packet-Forwarding Game
  - Reputation-Based Cooperation Mechanisms
    - CONFIDANT
    - CORE
Summary

- Two major TRM categories:
  - Individual-level trust
  - System-level trust

- Refer to Yu et al. survey paper for more detailed info
A Survey of Attack and Defense Techniques for Trust and Reputation Systems

- **Formulation**: the mathematical back-end to calculating trust/reputation metrics
- **Calculation**: the specific algorithm used to calculate the metrics
- **Dissemination**: the mechanism that allows the participants in the network to receive the metrics calculated
Attacker Model

• Attacker is an insider
  ◦ Legitimate access to the system
  ◦ Can behave appropriately

• Motivated by selfish or malicious intent

• Works alone or in a coalition

• Attacker is active
Attack Classification

• **Self-Promoting**: attacker falsely increases own reputation

• **Whitewashing**: attacker dynamically repairs their reputation after performing mischief

• **Slandering**: attackers falsely lower the reputation of reputable nodes

• **Orchestrated**: aggregated attack, using the above methods

• **Denial of Service**: DoS to prevent the calculation and dissemination of reputation values
Self-Promoting Attacks

**Assumptions:** (i) *positive feedback* must be a component in the trust calculation, (ii) the exploit attacks the trust formulation

**Vector description:**

- Fabricate fake positive feedback about self or modify own reputation during the dissemination process promote yourself at everyone else's expense

**Mitigation:** require accountability, require proof of successful transactions, prevent multiple identities
Whitewashing Attacks

**Assumptions:** negative feedback must be a component in the trust calculation

**Vector description:**

- Abuse the system, let trust metric degrade, and then use a vulnerability in the system to refresh their reputation (like re-entering the system under a new identity)
- More effective when combined with other attacks (like slandering)

**Mitigation:** redesign system to have different metrics for newcomers, takes into account limited history, limits user from re-establishing a new identity too quickly
Slandering Attacks

**Assumptions:** (i) *positive feedback* must be a component in the trust calculation, (ii) the exploit attacks the trust formulation dimension, (iii) usually performed by a coalition of attack nodes

**Vector description:**

- One or more identities produce negative feedback about other identities
- Predicated on the *lack of authentication* and *high sensitivity to negative feedback*

**Mitigation:** compute reputation based exclusively on direct information*, limit number of node identities, etc.
Orchestrated Attacks

**Assumptions:** multiple nodes collude

**Vector description:**

- Multiple, malicious nodes perform attacks on different attack vectors
- *Oscillation Attack:* colluders divide into teams, one team boosts reputation, other slanders, and switch after a period of time
- Fundamentally targets a system's formulation

**Mitigation:** really difficult to recognize collusion - attempt to identify nodes working together
Denial of Service (DoS) Attacks

**Assumptions:** (i) you are a malicious, nonrational attacker, (ii) the exploit attacks the trust formulation dimension, (iii) usually performed by a coalition of attack nodes

**Vector description:** subvert the entire reputation system, by overloading the its network or computation resources

**Mitigation:** implement enough of redundancy, don't automate the reputation system
Defense Strategies

Preventing Multiple Identities

• Mitigation Generation of False Rumors
• Mitigation Spreading of False Rumors
• Preventing Short-Term Abuse of the System
• Mitigating Denial of Service Attacks
Problem:

• In systems where generating new identities incurs little to no cost, malicious users can provide phantom feedback to the system.

• The identities can be generated/verified by a central authority (in a centralized system) or by their peers (in a decentralized system).
Preventing Multiple Identities (cont'd)

Solution:

• Centralized Approach
  ◦ An authority is responsible for issuing and verifying credentials for new nodes
  ◦ Provide new identity at a cost (computational or monetary)
  ◦ Introduces a single point of failure to the system

• Decentralized Approach
  ◦ Binding IP address to issued public key
  ◦ Use social knowledge of the network to generate and propagate a "web of trust"
Mitigation Generation of False Rumors

**Problem:**

- Attackers may generate false rumors by fabrication or modification of system reputation calculation

**Solution (#1), for selfish attackers:**

- Implement irrefutable proofs and digital signatures
  - Often implemented using cryptographic mechanisms
  - All feedback must be associated with proof of a valid transaction (the interaction with two identities) within the system
Mitigation Generation of False Rumors (cont'd)

Solution (#2), for reasonably small coalitions:

- Utilize dishonest feedback filter, using similarity measure to rate the credibility of reported feedback
  - Case #1: feedback is similar to first-hand experience and other received feedback -- *use in reputation calculation*
  - Case #2: feedback is too dissimilar -- drop and/or initiate counter-attack
Mitigation Spreading of False Rumors

Problem:
• Not every pair of interactions is symmetric

Solution (#1):
• Rely on pre-trusted identities to reduce the effectiveness of fabricated or altered information
  ◦ Poses a huge problem if one such node is actually malicious
    ▪ Employ (manual) integrity check mechanisms
    ▪ Check & balances on pre-trusted nodes
    ▪ Never allow pre-trusted nodes to be trusted absolutely
Mitigation Spreading of False Rumors (cont'd)

Solution (#2):

- Employ statistical and systems methods for building robust formulations that can be reasoned about in a precise fashion, i.e.,
  - Bayesian framework -- probability of a node misbehaving is modeled as a function of a Beta distribution, with loop feedback for the parameters
  - Feedback controls applications -- real-time adjustment of historical information in the calculation of the local reputation value
Preventing Short Term Abuse of the System

**Problem:**

- Malicious nodes will abuse the system, lowering the their reputation, and then behave appropriately for a period to improve their reputation
- New nodes are assigned the same level of trust as trustworthy nodes

**Solution:**

- Require new nodes to improve their level of reputation before being trusted
- New nodes must "pay their dues" and provide more service than they receive for a period of time
- Provide larger weights on recent negative behavior
Mitigating Denial of Service Attacks

**Problem:**

Denial of Service attacks on the dissemination system can prevent trust metrics from being sent to nodes, allowing malicious nodes to operate without detection.

**Solution:**

Randomize the selection of the node responsible for the calculation and dissemination of the trust metrics.

Employ common routing DoS prevention techniques.

Use of acknowledgements, multi-path dissemination, gossip mechanisms, and forward error-correction codes.
What about Ad-Hoc networks?

- Typical routing protocols for ad-hoc networks assume that all nodes will behave properly.
- Secure routing protocols based on cryptography require a trusted third party to handle key exchange and thus are not well-suited to ad-hoc networks.
- Pirzada et al propose a solution to this problem managing trust based on human behavioral models.
Human Behavioral Models

• The basic ideas behind this concept are that trust increases as a result of good behavior and decreases as a result of bad behavior, and that a trusted party can vouch for an unknown third party.
How is this implemented?

- Dynamic Source Routing (DSR) - Routes generated dynamically, but each packet follows a strict path.
- Each node maintains its own trust map for the other nodes in the network. This trust information is then shared with other nodes in the network.
- The trust information is then used by individual nodes to compute the most trustworthy route.
The Trust Model

- “Effort-Return” based trust model.
- Nodes maintain a trust counter for neighboring nodes.
- The counter is increased when a node forwards packets correctly, and decreased when a node tampers with a packet or fails to forward it.
- Let's look at a simple example of this.
The Trust Model

- From the perspective of the blue node:
  - Initially the trust values of all 4 neighboring nodes are zero.
  - The node then attempts to send packets to distant nodes (not shown) using the yellow nodes as an intermediate step.
  - The blue node then listens to see what the yellow nodes do.
The Trust Model

- Here, two of the nodes have properly retransmitted the packet, one didn't retransmit, and another tampered with the packet.

- The blue node will update its trust map based on these values, and will prefer using the nodes with higher trust values to send packets in the future.
The trust model provides for assigning different weights to the two components of the trust value (forwarding and accuracy).

- In this case, tampering with packets is considered to be a less desirable behavior than not forwarding at all.
- This local trust data is shared with other nodes by inserting it into transmitted packets (the direct trust value of the preceding node is inserted by the current node). Thus, other nodes in the network can 'learn' trust values from nodes they choose to trust.
EigenTrust

- Motivation – Filtering out fake content in P2P filesharing networks.
- End users rank transactions as positive or negative. This information is automatically traded between peers to develop transitive trust.
A user searches for a file, and gets several results that appear to be what she wants.
A user searches for a file, and gets several results that appear to be what she wants.

The user downloads from a peer.
A user searches for a file, and gets several results that appear to be what she wants.

The user downloads from a peer.

This is not the file that was requested, so this peer will receive a negative vote.
**EigenTrust Details**

- Nodes create local trust values for nodes based on direct experience. They then request trust values for other nodes from those nodes, and weight them according to their direct trust.

- By using this information from neighbors, a node can build a trust map of the network.

- Trust is disseminated through a Distributed Hash Table (DHT). A node, the *score manager* is responsible for calculating, storing, and disseminating reputation values for some subset of the network.
EigenTrust Security & Issues

• Malicious Score Manager behavior is ameliorated by randomly assigning multiple SMs to each node.

• Problem – EigenTrust doesn't give any preference to new trust data. Therefore, an attacker could distribute legitimate files for a period of time and then begin distributing undesirable content.
Other ideas in P2P Filesharing

- **Scrivener** – Based on economics. Credit/debit system where one *spends* credits when downloading files from peers and *gains* credits when serving files to peers.

- **P2Prep** – Uses fuzzy logic to aggregate user opinions and generate trust. Trust based on an entity's reputation and environmental data such as the last modification time.
CORE

- Users classified as "service providers" or "service requesters"
- Provider only accepts requests from requesters with trust higher than threshold
  - Provider can warn other nodes of DoS attacks
- Requester, when denied, makes a record of the negative behavior
- CORE only lets positive testimonies propagate
- Eliminates badmouthing attacks
Questions, Anyone?
References

- A Survey of Trust and Reputation Management Systems in Wireless Communications

- Trust-based Routing for Ad-Hoc Wireless Networks
  - [Pirzada et al., ICON 2004]

- A Survey of Attack and Defense Techniques for Reputation Systems
  - [Hoffman et al.]