Secure Routing in MANETS without Verifiable Distances or Link State

J.J. Garcia-Luna-Aceves

University of California Santa Cruz (UCSC)
jj@soe.ucsc.edu

http://www.cse.ucsc.edu/research/ccrg/home.html
Research Output Since Last Meeting

- Rolando Menchaca-Mendez, Ph.D. in Computer Engineering. Now an Assistant Professor at IPN, Mexico.

- Three papers published since last review meeting:

- More than two papers under submission; three invention submissions related to time ordering.
Outline

- Review of progress on time ordering for on-demand routing and secure routing
- Summary of research directions on multicasting, proactive routing and cross-layering
Two Levels of Security

- **Routing Plane**
  - Is the ordering of routing correct?
  - Do paths lead to the destination they claim?

- **Data Plane**
  - Are data packets being delivered?
  - Are data packets being corrupted?
Routing Plane Problem: Adversaries Lie

- Traditionally, nodes report their distance/link state to their neighbors

- Introduces a point of weakness in the routing protocol
  - Nodes can misrepresent their relative location in the network
    - Force traffic to flow through them
      - Perform attacks on the data
Past Approaches to Securing the Routing Plane

- Identify and eliminate adversaries
  - Can take time to correctly identify attacks
  - Packet loss can be due to poor routing behavior

- Attempt to secure the routing metric, e.g., use of hash chains
  - Computationally intensive
Secure Time Ordered Protocol for Routing

- A secured routing protocol for MANETs
  - Secures both the control plane and the data plane

- Three components:
  - Time based ordering using local relative times
  - End-to-end feedback
  - Path diversity

- Results:
  - Attackers cannot distort the topology in their favor
  - Attacks on signaling and data packets are detected
  - STOP is capable of correcting malicious behavior
Orderings Using Time

- **Temporal orderings are:**
  - Continuous
    - Unique values
  - Non-decreasing
    - Basis for loop-free routing
    - No need for sequence numbers, which means no network-wide resets!
  - Implicit
    - Secure

- **Definition:**
  - A is a *successor* of B to destination D if $T^A > T^B + \delta$
  - If B is a successor of A to destination D, then A is a *predecessor* of B to D.
Time Based Ordering: Example

- **RREQ**, states interest:
  - $t + \delta + \Delta_A$
  - $t + \delta + \Delta_B$
  - $t + \delta + \Delta_F$
  - $t + \delta + \Delta_G$

- **RREP**, states existence:
  - $t + 2\delta + \Delta_C + \Delta_B$
  - $t + 3\delta + \Delta_I + \Delta_H + \Delta_G$
  - $t + 2\delta + \Delta_H + \Delta_G$

RREQs label nodes with local times when they are sent & received.
RREPs propagate along loop-free paths based on those labels.
Time Based Ordering: Example

Why it works:
Local time at each node can only increase; hence, each node can order itself relative to its neighbors.
Time can only increase; hence, hop-by-hop orderings define a DAG.
Playing with Local Times

- Basic time ordering *may* render too few successors for a destination at a given node!
- Nodes can balance the percentage of predecessors and successors by changing the $\Delta$ parameter; the longer the value the fewer the successors a node perceives.
- **Approach:** Play with time as we play with congestion avoidance in end-to-end protocols.
  - If a node had more successors that predecessors in the previous ordering, it multiplicatively increases the value of $\Delta$ such that $\Delta = 1.1 \times \Delta$.
  - If the situation is reversed, the value of $\Delta$ is multiplicatively decreased such that $\Delta = 0.9 \times \Delta$.
  - The value of $\Delta$ is bounded between 0.1ms and 0.6ms, using a modulo operation, to ensure the value does not become too large so as to hinder the performance of the network.
Routing protocols based on spatial ordering and sequence numbers require global resets.

- Trust is on the value of the sequence number.

Example: Link state algorithm uses sequence number and age for each link. Sequence number is increased up to a maximum and each value is aged out. Source of link state must refresh the link state periodically. Large sequence number space must be used.

With time ordering, a flood ID is taken from a non-sequential space, and the only requirement is that the source not use the same number more than once to identify a flooding. No need to age out and refresh states.

- Trust is on relative time when signaling packets arrive or are sent, and flood ID simply prevents duplicate propagation.
Security of Temporal Orderings

- Goal is to have ordering not depend on what nodes say, but rather when they say it.
- RREQ and RREP are signed by sender and state source node ID, destination node ID, flood ID.
- No intermediate node can change a control packet.
- No need for network-wide rest of sequence numbers.
- Nodes cannot manipulate time in their favor!
  - A node can only delay the forwarding of RREQs and RREPs; this makes it less attractive as a successor to destinations.

Temporal orderings are therefore less vulnerable to misrepresentations than explicit statements of distances or link states.
Detection and Correction

- Digital signatures are used to guarantee authenticity of RREQs and RREPs.
- Time-based ordering ensures the integrity of route computation.
- Flood IDs are used to avoid unnecessary transmissions; however, they are not sequence numbers (e.g., sequence 10, 2000, 5, ... is valid).
- Multiple (two or more) paths used to route data
  - Help increase the probability of finding a secured path.
- End-to-end feedback used to determine the proportion of packets delivered along each path
  - This will detect if an attacker along one path is dropping or modifying data packets.
- Load balancing used to vary the amount of packets sent along each path
  - Decision based on feedback
  - A blacklist flag in RREQs can force a different ordering of the nodes when necessary.
Detection and Correction, Example

- S-D are not adversaries, digital signatures cannot be forged, there is some path from S to D without attackers, and RREQ/RREP state node ID, destination node ID, flood ID.

- Attacks to route discovery process:
  - Delete routes, drop RREQ, RREP: Use multiple paths.
  - Fabricate RREQ, RREP: Use signed RREQ, RREP.
  - Replay RREQ, RREP: Use the flood ID; replays look less attractive.
  - Modification of RREQ or RREP: No entry can be modified by relays.
  - Delaying forwarding of RREQ, RREP: Attacker is less attractive.
Performance Results (in submission)

- 100 nodes
- 10 flows, 800 packets each 4 packets per second
- Random waypoint, speed of 1-10mps and 30 second pause time
- 1000m x 1000m area, with 150m transmission range
- 900 seconds, repeated 50 times.
- Attackers change hop count, colluders tunnel control packets

### No Attackers

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Delivery ratio</th>
<th>Delay</th>
<th>Overhead</th>
</tr>
</thead>
<tbody>
<tr>
<td>AODV</td>
<td>0.60 +/- 0.10</td>
<td>0.09 +/- 0.04</td>
<td>14.4 +/- 5.3</td>
</tr>
<tr>
<td>DSR</td>
<td>0.14 +/- 18.5</td>
<td>18.5 +/- 15.9</td>
<td>5.0 +/- 1.2</td>
</tr>
<tr>
<td>OLSR</td>
<td>0.30 +/- 0.08</td>
<td>0.07 +/- 0.02</td>
<td>67.5 +/- 1.2</td>
</tr>
<tr>
<td>ARAN</td>
<td>0.53 +/- 0.09</td>
<td>0.21 +/- 0.11</td>
<td>24.7 +/- 5.0</td>
</tr>
<tr>
<td>STOP</td>
<td>0.71 +/- 0.11</td>
<td>0.07 +/- 0.02</td>
<td>17.6 +/- 2.2</td>
</tr>
</tbody>
</table>

### Independent Attackers

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Delivery ratio</th>
<th>Delay</th>
<th>Overhead</th>
</tr>
</thead>
<tbody>
<tr>
<td>AODV</td>
<td>0.29 +/- 0.11</td>
<td>0.03 +/- 0.01</td>
<td>8.8 +/- 2.2</td>
</tr>
<tr>
<td>ARAN</td>
<td>0.33 +/- 0.12</td>
<td>0.26 +/- 0.11</td>
<td>10.2 +/- 1.5</td>
</tr>
<tr>
<td>STOP</td>
<td>0.38 +/- 0.09</td>
<td>0.05 +/- 0.02</td>
<td>12.8 +/- 4.5</td>
</tr>
</tbody>
</table>

40% of the nodes are adversaries.
- 10% modify hop count
- 10% drop RREP
- 10% drop data packets
- 10% delay control packets
Performance Results (MASS 2010)

### TABLE I
**SIMULATION PARAMETERS**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation Time</td>
<td>900s</td>
</tr>
<tr>
<td>Number of Nodes</td>
<td>100</td>
</tr>
<tr>
<td>Simulation Area</td>
<td>1000m x 1000m</td>
</tr>
<tr>
<td>Node Placement</td>
<td>Uniform</td>
</tr>
<tr>
<td>Mobility Model</td>
<td>Random Waypoint</td>
</tr>
<tr>
<td>Min-Max Speed</td>
<td>1-10m/s</td>
</tr>
<tr>
<td>Pause Time</td>
<td>30s</td>
</tr>
<tr>
<td>Propagation Model</td>
<td>Two-ray</td>
</tr>
<tr>
<td>Physical Layer</td>
<td>802.11</td>
</tr>
<tr>
<td>Antenna Model</td>
<td>Omnidirectional</td>
</tr>
<tr>
<td>MAC Protocol</td>
<td>802.11 DCF</td>
</tr>
<tr>
<td>Data Source</td>
<td>CBR</td>
</tr>
<tr>
<td>Number of Packets per Flow</td>
<td>400</td>
</tr>
<tr>
<td>Packet Rate</td>
<td>4 packets per second</td>
</tr>
<tr>
<td>Number of Flows</td>
<td>50</td>
</tr>
</tbody>
</table>

### TABLE II
**VARIED SIMULATION PARAMETERS**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Scenario A</th>
<th>Scenario B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Nodes</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Simulation Area</td>
<td>1000x1000m</td>
<td>1000x1000m</td>
</tr>
<tr>
<td>Radio Range</td>
<td>150m</td>
<td>200m</td>
</tr>
</tbody>
</table>

### TABLE III
**SIMULATION RESULTS**

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Delivery Ratio</th>
<th>Latency</th>
<th>Overhead</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Scenario A</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AODV</td>
<td>0.58±0.05</td>
<td>0.089±0.015</td>
<td>17.8±4.5</td>
</tr>
<tr>
<td>OLSR</td>
<td>0.10±0.05</td>
<td>0.031±0.006</td>
<td>16.3±0.2</td>
</tr>
<tr>
<td>ARAN</td>
<td>0.48±0.07</td>
<td>0.31±0.08</td>
<td>22.2±5.0</td>
</tr>
<tr>
<td>CaSH</td>
<td>0.71±0.06</td>
<td>0.17±0.065</td>
<td>7.1±1.7</td>
</tr>
<tr>
<td>TORP</td>
<td>0.80±0.07</td>
<td>0.11±0.02</td>
<td>7.1±2.5</td>
</tr>
<tr>
<td>DYSO</td>
<td>0.38±0.09</td>
<td>0.24±0.07</td>
<td>5.53±0.9</td>
</tr>
<tr>
<td>DSR</td>
<td>0.05±0.02</td>
<td>0.99±0.3</td>
<td>3.4±0.23</td>
</tr>
<tr>
<td><strong>Scenario B</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AODV</td>
<td>0.78±0.05</td>
<td>0.067±0.015</td>
<td>7.9±1.9</td>
</tr>
<tr>
<td>OLSR</td>
<td>0.23±0.03</td>
<td>0.034±0.004</td>
<td>12.6±0.2</td>
</tr>
<tr>
<td>ARAN</td>
<td>0.82±0.07</td>
<td>0.17±0.05</td>
<td>13.4±4.0</td>
</tr>
<tr>
<td>CaSH</td>
<td>0.93±0.05</td>
<td>0.064±0.022</td>
<td>4.0±0.46</td>
</tr>
<tr>
<td>TORP</td>
<td>0.95±0.03</td>
<td>0.073±0.011</td>
<td>2.13±0.6</td>
</tr>
<tr>
<td>DYSO</td>
<td>0.53±0.07</td>
<td>0.24±0.08</td>
<td>5.2±1.0</td>
</tr>
<tr>
<td>DSR</td>
<td>0.06±0.04</td>
<td>1.2±0.7</td>
<td>3.6±0.5</td>
</tr>
</tbody>
</table>

Time ordering is a good thing even w/o security benefits of not using distances or link states.
Next Steps and Questions

- Demonstrate whether time ordering can/should be used in proactive routing protocols (distance vectors and link states)
- Complete our results on secure multicasting with and without anonymous receivers.
  - Multicast results expected by early Fall quarter.
- Address cross-layering with MAC protocol (tbd)
- Key problem to address in the future:
  - The role of end-to-end probing in multicast groups.
Time Ordering for Proactive Routing

- The main challenge is that routers need to combine multiple advertisements to make their signaling more efficient (send k updates in a message rather than k messages with one update each).

- The basis of the solution is to use signaling packets that do not contain any piece of information that must be processed over multiple hops (e.g., a distance).

- An update states:
  
  destination ID, [next hop IDs], update ID, reset flag
  
  - Plus auxiliary information (e.g., distance statement or link statement not used for ordering)
  
  - The destination of an update can be a node or a link [stated by head and tail nodes], or something else

Details at our next meeting!
Example of Time Ordering for Proactive Routing

- Update states existence
- Updates propagate along loop-free paths dictated by relative times.

\[
t + \delta + \Delta_A \\
t + \delta + \Delta_B \\
t + 2\delta + \Delta_C + \Delta_B \\
t + 3\delta + \Delta_I + \Delta_H + \Delta_G \\
t + \delta + \Delta_F \\
t + \delta + \Delta_G \\
t + 2\delta + \Delta_H + \Delta_G
\]
Example of Time Ordering for Proactive Routing

Time ordering can be used very efficiently instead of spatial ordering in proactive routing (link state or distance vectors).

### Simulation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Nodes</td>
<td>100</td>
</tr>
<tr>
<td>Number of flows</td>
<td>50</td>
</tr>
<tr>
<td>Number of packets per flow</td>
<td>400</td>
</tr>
<tr>
<td>Flow rate</td>
<td>2 packets per second</td>
</tr>
<tr>
<td>Number of link faults</td>
<td>25% of number of links</td>
</tr>
<tr>
<td>Simulation time</td>
<td>900 seconds</td>
</tr>
</tbody>
</table>
Sample Simulation Results

- Used **DUAL (Cisco’s EIGRP)** for testing; link state comparison would be too easy (no periodic flooding with time ordering!)

<table>
<thead>
<tr>
<th>8 neighbors per node</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Delivery Ratio</td>
<td>Average end-to-end delay</td>
<td>Total overhead packets per flow</td>
</tr>
<tr>
<td>Hybrid DUAL</td>
<td>0.94 +/- 0.07</td>
<td>0.24 +/- 0.01</td>
<td>326 +/- 47</td>
</tr>
<tr>
<td>Temporal DUAL</td>
<td>0.91 +/- 0.06</td>
<td>0.24 +/- 0.02</td>
<td>356 +/- 55</td>
</tr>
<tr>
<td>DUAL</td>
<td>0.80 +/- 0.08</td>
<td>0.24 +/- 0.01</td>
<td>552 +/- 89</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4 neighbors per node</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Delivery Ratio</td>
<td>Average end-to-end delay</td>
<td>Total overhead packets per flow</td>
</tr>
<tr>
<td>Hybrid DUAL</td>
<td>0.87 +/- 0.08</td>
<td>0.37 +/- 0.02</td>
<td>181 +/- 27</td>
</tr>
<tr>
<td>Temporal DUAL</td>
<td>0.91 +/- 0.05</td>
<td>0.36 +/- 0.01</td>
<td>177 +/- 21</td>
</tr>
<tr>
<td>DUAL</td>
<td>0.82 +/- 0.04</td>
<td>0.36 +/- 0.04</td>
<td>231 +/- 25</td>
</tr>
</tbody>
</table>

- Same delivery ratio and delays with much less overhead in wired network; advantage increases with density.
Secure Time-Ordered Multicasting Meshes and Enclaves
Secure Time Ordered Multicasting

First approach to ensure correct routing along shared multicast meshes without the use of explicit distance or link-state information.

- Each core of a multicast group sends announcements (core ID, group ID, flood ID) to time order nodes in the MANET with respect to the core.
- Each receiver sends join requests (receiver ID, group ID, request ID) towards core along one or more neighbors. Nodes on the multicast mesh send join replies (receiver ID, group ID, request ID).
- Any chosen relay for the multicast group must be between the core and a receiver in the time ordered multicast mesh.
- A source selects a receiver towards the core; the first relay in the multicast mesh starts flooding the packet in the mesh.
- Packet caches used to avoid packet duplicates.
- With anonymous receivers, some scalable probing must be used to protect the data plane.
Probing in Time Ordered Multicast

- Similar to having multiple unicast S-D enclaves probed.
- Probing ensures that data plane behaves properly.
- Simple for a few receivers; what about the case of very many receivers?
Conclusion

- Temporal ordering is a new approach to developing more efficient routing protocols and to securing more effectively the ordering of routing protocols in wired and wireless networks.

- Temporal ordering can be applied to unicast and multicast routing, proactive and reactive routing.

- Next steps:
  - Complete the work on proactive time-ordered routing
  - Complete time-ordered multicast routing
  - Handling of ``end-to-end’’ feedback still needs more thought.
  - Handling malicious use of valid node IDs.
Thanks!