Secure routing in IoT networks with SISLOF

Ayman El Hajjar\textsuperscript{1,*}, George Roussos\textsuperscript{1}, Maura Paterson\textsuperscript{2}

\textsuperscript{1} Department of Computer science and Information systems
\textsuperscript{2} Department of Economics, Mathematics and Statistics
Birkbeck University of London, UK
a.elhajjar@bbk.ac.uk - @azelhajjar

June 8, 2017

Birkbeck
UNIVERSITY OF LONDON
Outline

- Litterature Review
- Routing protocol for Low Power and lossy networks (RPL)
- Shared Identifier Secure Link Objective Function (SISLOF) Rationale
- SISLOF Objective Function: How does it work
- Message and Modifications
- TestBed: Experiment Design
- TestBed: Experiment parameters
- SISLOF Rationale (Previous experiments Results)
- SISLOF Performance Results
- Summary and Further Work
Litterature Review

1 Internet of Things networks properties are similar to DSN and WSN
2 DSN and WSN Security with Eschenauer and Gligor key pre-distribution scheme:
   - Probability of 50% for nodes \(^1\) in the network to share keys is enough to guarantee full secure network connectivity.
3 Conventional routing protocols are not suitable for the Internet of Things.
4 The Routing Protocol for Low-Power and Lossy Networks (RPL) is a distance vector IPv6 routing protocol optimized for the IoT networks.
   - RPL organises its topology in a Directed Acyclic Graph (DAG).
   - Only nodes that are in the DAG can communicate with each other.

\(^1\)Stirling approximation

A.ElHajjar, G.Roussos, M.Paterson - @azelhajjar

Securing IoT with SISLOF
Routing protocol for Low Power and lossy networks (RPL)

Figure: Mote 4 sends a DIO multicast message to all neighbours (candidate parents)

Figure: All motes that received the DIO message reply with a unicast DAO message

Figure: Mote 4 decides on which mote will be the preferred parent

RPL routing table formation. Mote 4 choosing a preferred parent.
Many of the available standards and protocols for conventional IP based networks are not suitable for the internet of Things networks.

- Eschenauer and Gligor key pre-distribution algorithm on IoT networks does not achieve full connectivity when applied on IoT networks using RPL and same ring sizes as DSN.
Many of the available standards and protocols for conventional IP based networks are not suitable for the internet of Things networks.

- Eschenauer and Gligor key pre-distribution algorithm on IoT networks does not achieve full connectivity when applied on IoT networks using RPL and same ring sizes as DSN.
- To achieve full connectivity using the Eschenauer and Gligor keys pre-distribution Algorithm, much larger key rings were needed.
Many of the available standards and protocols for conventional IP based networks are not suitable for the internet of Things networks. 

- Eschenauer and Gligor key pre-distribution algorithm on IoT networks does not achieve full connectivity when applied on IoT networks using RPL and same ring sizes as DSN.
- To achieve full connectivity using the Eschenauer and Gligor keys pre-distribution Algorithm, much larger key rings were needed.
  - For a 100,000 motes network a 4,600 keys in the ring will be needed to achieve full connectivity.
Shared Identifier Secure Link Objective Function (SISLOF) Rationale

- Many of the available standards and protocols for conventional IP based networks are not suitable for the internet of Things networks
  - Eschenauer and Gligor key pre-distribution algorithm on IoT networks does not achieve full connectivity when applied on IoT networks using RPL and same ring sizes as DSN.
  - To achieve full connectivity using the Eschenauer and Gligor keys pre-distribution Algorithm, much larger key rings were needed.
    - For a 100,000 motes network 4,600 keys in the ring will be needed to achieve full connectivity.
    - Each mote has 90 kb of memory storage.
Many of the available standards and protocols for conventional IP based networks are not suitable for the internet of Things networks.

- Eschenauer and Gligor key pre-distribution algorithm on IoT networks does not achieve full connectivity when applied on IoT networks using RPL and same ring sizes as DSN.
- To achieve full connectivity using the Eschenauer and Gligor keys pre-distribution Algorithm, much larger key rings were needed.
  - For a 100,000 motes network a 4,600 keys in the ring will be needed to achieve full connectivity.
  - Each mote has 90 kb of memory storage.
  - 54 kb of this will be used for rings storage (identifiers and keys).
Many of the available standards and protocols for conventional IP based networks are not suitable for the internet of Things networks.

- Eschenauer and Gligor key pre-distribution algorithm on IoT networks does not achieve full connectivity when applied on IoT networks using RPL and same ring sizes as DSN.

- To achieve full connectivity using the Eschenauer and Gligor keys pre-distribution Algorithm, much larger key rings were needed.

  - For a 100,000 motes network a 4,600 keys in the ring will be needed to achieve full connectivity.
  - Each mote has 90 kb of memory storage.
  - 54 kb of this will be used for rings storage (identifiers and keys).
  - Motes took on average 23 seconds to compute and compare larger rings and used 87% of the processing power.
Many of the available standards and protocols for conventional IP based networks are not suitable for the internet of Things networks.

- Eschenauer and Gligor key pre-distribution algorithm on IoT networks does not achieve full connectivity when applied on IoT networks using RPL and same ring sizes as DSN.
- To achieve full connectivity using the Eschenauer and Gligor keys pre-distribution Algorithm, much larger key rings were needed.
  - For a 100,000 motes network a 4,600 keys in the ring will be needed to achieve full connectivity.
  - Each mote has 90 kb of memory storage.
  - 54 kb of this will be used for rings storage (identifiers and keys).
  - Motes took on average 23 seconds to compute and compare larger rings and used 87% of the processing power.
- Eschenauer and Gligor key pre-distribution algorithm on IoT is not feasible without any modification.
How does SISLOF achieve this:

1. Motes select random rings (keys and identifiers)
2. DIO messages send downward to all neighbours.
   - The number of DIO messages defer depending on the number of identifiers in the ring.
   - Each time a mote receive a DIO message, it consider the originator of the message a "candidate parent"
   - It compares its own identifier ring with the identifier ring embedded in DIO message
3. For each DIO message, the receiver mote replies back with a DAO message informing the sender if they will be chosen as a preferred parent
   - If yes: Which identifier they have in common?
   - If no: The mote cannot be a preferred parent
     - No shared identifier
     - Shared identifier exist but another more will be chosen as preferred parent.
Message and Modifications

- Addition to the DODAG Information Object (DIO) message:
  - 1 byte for each of the variables
    - Ring Size (RS)
    - Identifier size (b)
    - Number of identifiers in one message (NI)
    - Number of Sequence (NS)
    - Sequence Number (SN)
  - ID SN for the number of identifiers sent in the message.
    - 33 bytes in the payload remain for sending identifiers from the ring.

---

 Addition to the DIO message

```
<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS</td>
<td>b</td>
<td>NS</td>
<td>NI</td>
<td>ID_{SN}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

---

A.ElHajjar, G.Roussos, M.Paterson - @azelhajjar

Securing IoT with SISLOF
Addition to the Destination Advertisement message (DAO) message:

- 1 byte for Sequence Number (SN)
- 1 byte for Number of identifiers in one message (NI) where the bitmap representing shared identifiers bits.

Addition to the DAO message.
TestBed: Experiment Design

**Independent variables**
- Pool size & Number of motes (100, 250, 500, 750, 1000, 2500)
- Ring size (8, 13, 18, 22, 25, 41)

**Control Variables**
- 64 bits key.
- 32 bits identifier.
- $250m^2$ simulation area - similar to the university campus area.
- 5 runs for results consistency.
TestBed: Experiment Parameters

Experiment Platform
- Zolertia Z1 motes were used.

Contiki
The Open Source OS for the Internet of Things
TestBed: Experiment Parameters

**Experiment Platform**

- Zolertia Z1 motes were used.
  - 90 Kb memory storage
  - 50 meters transmitting range
- Contiki OS for the 6LoWPAN stack (RPL, CoAP, ContikiMAC).
- Cooja Simulator

Zolertia Z1 low-power wireless module for IoT and WSN
SISLOF Rationale (Previous experiments Results)

1. Key pre-distributed for DSN on IoT
2. Larger key ring size to achieve full connectivity

<table>
<thead>
<tr>
<th>Pool Size</th>
<th>DSN</th>
<th>IoT</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>8</td>
<td>23</td>
</tr>
<tr>
<td>250</td>
<td>13</td>
<td>36</td>
</tr>
<tr>
<td>500</td>
<td>18</td>
<td>48</td>
</tr>
<tr>
<td>750</td>
<td>22</td>
<td>63</td>
</tr>
<tr>
<td>1000</td>
<td>25</td>
<td>77</td>
</tr>
<tr>
<td>2500</td>
<td>41</td>
<td>104</td>
</tr>
</tbody>
</table>

DSN ring sizes vs. IoT ring sizes to achieve full connectivity.

Ring Size vs. % of Number of DAGs with a shared key until 100% is achieved.

A.ElHajjar, G.Roussos, M.Paterson - @azelhajjar

Securing IoT with SISLOF
All motes participate in the network in comparison with RPL using OF0 where note all motes.

Figure: Network Topology as seen for various implementation
Decrease in the number of keys/identifiers needed in the ring.

Table: Results Comparison: Number of motes N, Shared Keys SK (100% for IoT), Ring Size (RS)

<table>
<thead>
<tr>
<th></th>
<th>DSN</th>
<th>IoT OF0</th>
<th>IoT SISLOF</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>RS</td>
<td>SK %</td>
<td>RS</td>
</tr>
<tr>
<td>100</td>
<td>8</td>
<td>50.52</td>
<td>23</td>
</tr>
<tr>
<td>250</td>
<td>13</td>
<td>50.43</td>
<td>36</td>
</tr>
<tr>
<td>500</td>
<td>18</td>
<td>57.14</td>
<td>48</td>
</tr>
<tr>
<td>750</td>
<td>22</td>
<td>49.47</td>
<td>63</td>
</tr>
<tr>
<td>1000</td>
<td>25</td>
<td>57.14</td>
<td>77</td>
</tr>
<tr>
<td>2500</td>
<td>41</td>
<td>48.19</td>
<td>104</td>
</tr>
</tbody>
</table>

IoT SISLOF performance in comparison with DSN and IoT network (RPL with OF0)
Summary and Further Work

**Summary**

1. **Validated** that SISLOF provide full connectivity of the network while maintaining a smaller ring size for all motes within reach.
2. **Validated** that modifications of SISLOF do not add a large overhead on RPL messages.
3. **Convergence of secure routing table** occurred without the exchange of the keys.
4. **Decreased** the storage size of the ring for 100,000 motes network from 54Kb to 28.8 Kb.
5. **Keys** are not transmitted in any message. Only identifiers are transmitted.

**Further work**

- Investigate mobility impact on SISLOF
- Analyse overhead and changes SISLOF add to PRL in term of:
  - Number of hops
  - CPU usage
  - Duration for Routing table to converge
  - Number of exchanged control messages
Thank you