a framework for observing nodes’ behavior in IoT validation platforms

Orange Labs

Quentin Lampin, Dominique Barthel
Nov 6th 2014, “IoT workshop” at INRIA, Montbonnot
Agenda

part 1  a look back
part 2  SensOrLabs
part 3  an observability toolchain
part 4  some results
part 5  the way forward
A look back: project knitting

- Orange Labs working on WSN protocols since 2002
- ARESA1 demo (2008)
  - 86 nodes, full source code, in-band limited monitoring
- ARESA1 demo (2009)
  - 56 nodes, Coronis black-box radio, in-band limited monitoring
- Dec 2013 OLRE / ARESA2 demo
  - a 82-node fully observable platform (SensOrLabs)
  - out of band monitoring, systemic approach to observability
  - Coronis partnership
the SensOrLabs platform

- largely Senslab-inspired, but
  - rewrote infrastructure software
  - redesigned “gateway” board assembly
  - redesigned power monitoring board
  - hosts Elster/Coronis Excelyo nodes

- 82 nodes at Orange Labs Meylan
  - plugged into any site Ethernet socket
an observability toolchain

- a major hand-holding effort
- open node code instrumentation
- power monitoring processed similarly
- event timestamping
- PCAP encapsulation
- flow multiplexing

- Wireshark as a general event browsing and display tool
an underlying model for the network

- nodes, entities
an underlying model for the network (2)

- nodes, entities
- links
- packets

- already applied to very different comm. stacks
a taxonomy of observability events

<table>
<thead>
<tr>
<th>Event name</th>
<th>Hexadecimal value (1 byte)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EVENT_NODE_ADD</td>
<td>0x00</td>
</tr>
<tr>
<td>EVENT_NODE_PROPERTY_ADD</td>
<td>0x01</td>
</tr>
<tr>
<td>EVENT_NODE_PROPERTY_UPDATE</td>
<td>0x02</td>
</tr>
<tr>
<td>EVENT_NODE_REMOVE</td>
<td>0x03</td>
</tr>
<tr>
<td>EVENT_ENTITY_ADD</td>
<td>0x10</td>
</tr>
<tr>
<td>EVENT_ENTITY_PROPERTY_ADD</td>
<td>0x11</td>
</tr>
<tr>
<td>EVENT_ENTITY_PROPERTY_UPDATE</td>
<td>0x12</td>
</tr>
<tr>
<td>EVENT_ENTITY_REMOVE</td>
<td>0x13</td>
</tr>
</tbody>
</table>
a taxonomy of observability events (2)

- Wireshark dissectors remain totally generic
  - assuming the network model
Staged timestamping and PCAP encapsulation

- log time spent in event buffer
  - in open node clock ticks
- compute serialization time
- insert (adjusted) absolute time
- restore platform-wide time monotonicity

![Diagram showing the process of staged timestamping and PCAP encapsulation]

- Open Node (relative time)
- Gateway (absolute time)
- SensOrLabs server
- Reordering Muxing

- event
- serial
- NTP
Wireshark output
Orange Labs Network visualizer
The way forward

- Make observability event specifications public
- Submit Wireshark dissectors for commit

- Provide Open Source observability libraries
  - Contiki, TinyOS, RIOT, FreeRTOS
  - might use a little help on this one

- Version 2 of SensOrLab
  - “standard” open node interface
  - more generic gateways (e.g. Raspberry Pis)
  - GPS synchronization, enhanced timestamping/reordering
thank you