Intelligent Resilience in the Internet of Things

Edel Sherratt
The Challenge

• The IoT includes all manner of device, toy, safety critical element ...
• The IoT is neither contained nor constrained
• Failing elements are to be expected
• So too are hostile elements
• And early prototypes ...
Images of Aberystwyth Robotics Club
Test: can you find the serious science amongst the nonsense?

https://twitter.com/samtomindustrys
Environmental monitoring

What could possibly go wrong?
How about a smart fridge?
Or the famous SDL challenge?

- Tracks, public highway, gates, sensors, signals controller
- Cars on the road are part of the environment

http://www.sdl-forum.org/Events/SAM03Contest.htm
Unconstrained environments

• Area of ongoing research in robotics
• and computer vision
• and intrusion detection

The IoT is an unconstrained environment
Anomaly detection

• Key to ensuring resilience in an unconstrained environment
• Applied in robotics, vision, intrusion detection, industrial processes
• As well as wireless sensor networks
Training and Testing

• Labelled data is essential to train and test an anomaly detection system
• Getting good training data is problematic
  • Real data is noisy
  • giving non-identical distribution of training samples
• Published data sets are useful
• Keeping them up to date is challenging
Where SDL comes in

1. SDL+ as a method to create IoT systems with integral anomaly detection
2. SDL simulation as a source of high-quality bespoke training data
SDL+ Core Activities

SDL+ core activities

Analysis

Classified requirements
Named, defined concepts

Requirements capture

Requirements

Draft Design

Incomplete system coverage
Partly-formal specifications

Formalization

Formal SDL+ description
SDL+ core activities

• Analysis
  • Concepts with names and definitions form an ontology
  • Used to identify threats and propose countermeasures

• Design
  • Explore vulnerabilities associated with different designs
  • Explore options for anomaly detection

• Formalization
  • Include anomaly detection in the formal description
SDL model with external anomaly detection
External procedure classifies behaviour as normal or anomalous.
Design decision:

- include anomaly detection in the controller
- or distribute it across different system elements
SDL+ model is validated and tested
SDL+ testing and validation

• Both involve executing the SDL+ formal description
• Both use similar test cases
• Testing compares formal description with an implementation
• Validation compares formal description with classified requirements and with concepts from analysis
Validation

• Check syntax and context conditions
• Check that requirements are addressed
  • Represent different environmental conditions as combinations of events
  • TTCN-3, MSC, SDL-2010
• Execute the SDL+ description
Training data as a by-product of validation

• Validation results in execution traces
• Execution traces with signal payload constitute labelled training data
Use the results of validation to train anomaly detection subsystem(s)
Testing the anomaly detection subsystem

• Re-frame an established data set as events
• Test the SDL+ formal description, with its anomaly detection system
• Evaluate the resulting traces
But, so far, this is hypothetical

• Next step is to conduct some actual experiments
• For example, use the approach to re-create an existing IoT system, but this time with integral anomaly detection
• See how the resulting system behaves in the field
Further empirical work

• Evaluate different kinds of anomaly detection
• Discover what constitutes an acceptable level of false positives
• Explore different responses to anomalous situations