Combinatorial Testing Methodologies for IoT Hubs
Dimitris Simos, Bernhard Garn, Dominik Schreiber, Rick Kuhn, Raghu Kacker

Reducing Complexity for IoT Home Automation Systems with Combinatorial Methods

Research Problems
- The IoT industry is growing rapidly with estimations predicting that 1 million IoT devices are going to be purchased and installed each hour in 2021.
- IoT devices are often controlled by a hub, to integrate devices using different protocols, e.g., z-wave, zigBee, Wi-Fi.
- Exhaustive IoT testing quickly becomes intractable for larger systems, because of many devices interacting with each other.

Combinatorial Testing
- Reduce test cases while still achieving good coverage.
- With interaction strength 5-6, usually all software bugs can be found.
- Modelling of IoT home automation systems.
- Generation of test cases with combinatorial methods.
- Automatic test oracle to determine, if the test was successful.

Testing Methodologies
- Analysis of the structure of some IoT home automation hub systems.
- Home Assistant, OpenHab, Domoticz, Samsung Smart, Things, Philips Hue Hub.
- Developed two different test modelling methodologies, based on combinatorial modelling techniques.

Hub Automations Testing (HAT) Methodology
- Targets the automation component of the hub.
- In Home Assistant mostly accessible by custom scripting.
- Automations can have three parts (trigger, condition, action).

HUB Services Testing (HUST) Methodology
- Global perspective on the hub
- Modelling all devices connected to the hub and their exposed functionality

Case Study
- Case study on one concrete IoT home automation hub (Home Assistant).
- HUST methodology with 8 devices, HAT with 18 devices.
- Errors grouped in categories for combinatorial analysis.
- Both methodologies were able to find a variety of different errors in the expected behaviour.

Figures:
- Figure 1: Testing framework details
- Figure 2: Results for all error groups of HAT methodology
- Figure 3: Results for all error groups of HUST methodology
- Figure 4: Results for all error groups of HUST methodology