Conflict Verification in Connected IoT Apps

Aqsa Kashaf
Anand K. Prakash
Daniel Calderon

ConViCT

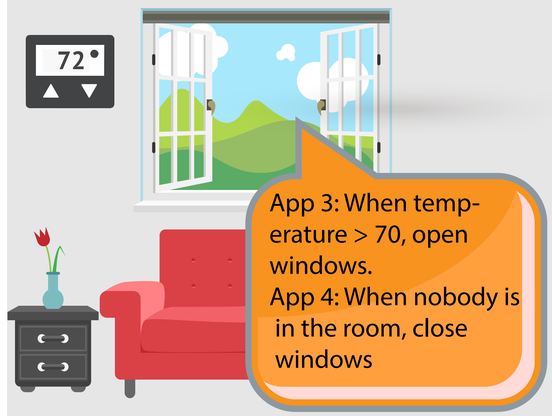
Motivation

User Specified IoT Apps can conflict leading to security Implications

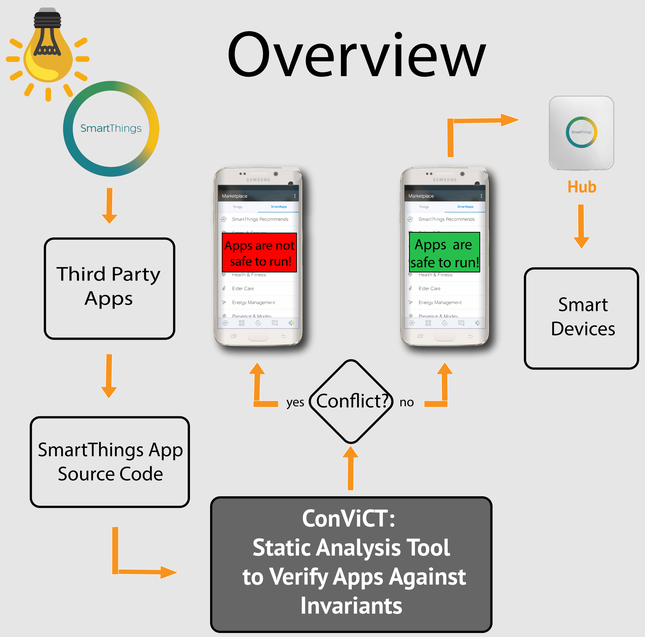
Conflicts

Direct Conflicts
Example: App 3 & 4

Indirect Conflicts
Example: App 1 & 2



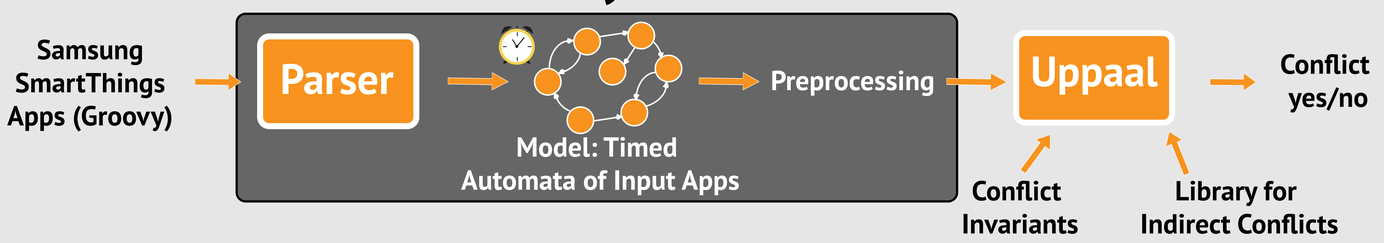
Overview



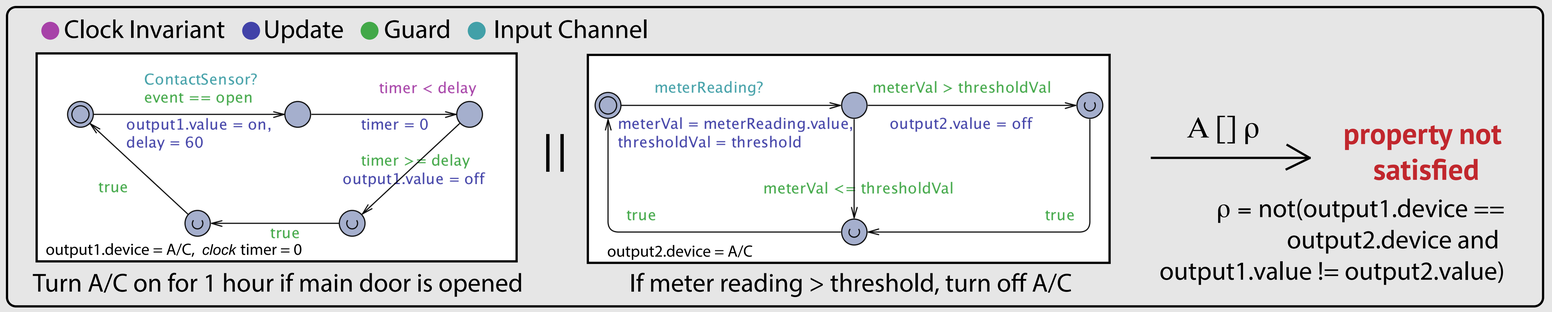
Challenges

- Complexity of IoT Apps
  1) Stateless 2) Discrete Stateful 3) Continuous Stateful (Time)
- Defining a unified model that covers the above mentioned categories of app, achieved by using timed automata.
- Parsing and Compilation of groovy code into timed automata.

System



Illustration



Turn A/C on for 1 hour if main door is opened

If meter reading > threshold, turn off A/C

The above figures show the parser output of two Samsung SmartThings apps with their description. This output is given to Upaal along with the conflict invariant described as ρ. Uppaal then composes the two automata in parallel and verifies the property.

Future Work

- Provide an API to users to define their own security invariants to verify against our tool.
- Develop a framework for conflict resolution.
- Evaluate the performance of the tool in terms of soundness and completeness.