A Brief Overview of Existing Tools for Testing the Internet-of-Things

João Pedro Dias, Flávio Couto, Ana C.R. Paiva and Hugo Sereno Ferreira

First International Workshop on Verification and Validation of Internet of Things (VVIoT)

9th of April 2018, Västerås - Sweden

U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

INESCTEC

Outline

- Introduction
- Research Challenges
- IoT Testing Solutions
- Comparative Overview
- Conclusion

Introduction

- Internet-of-Things relies on a combination of **hardware**, **software** and **architectures** that **enable real-world objects to sense and interact** with the surrounding **environment**, while being **Internet-connected and uniquely identifiable.**
- It is expected that *soon* more than 10 billion IoT devices will be connected.
- Systems are, by nature, error-prone. When systems are scaled up (complexity, features, number of devices, …), the number of errors increases with its scale.
- IoT systems are an example of such.

Introduction

Beyond the massive scale of IoT systems, other considerations must be taken into account:

- **Dynamic topologies**
- **Unreliable connectivity**
- **Device and protocols heterogeneity**

These characteristics lead to appearance of systems that are remarkably complex to test and validate (e.g. smart-homes, smart-cities,…).

Introduction

To guarantee IoT-based system's

- **performance, scalability, reliability, and security.**

It is needed focus on testing the different **layers and components** that make part of the system, from **low-level/hardware specifications to high-level components**.

IoT systems architecture can be sliced into *three layers*: **edge, fog and cloud.**

Each layer has different roles in the system, thus having different testing needs.

Introduction

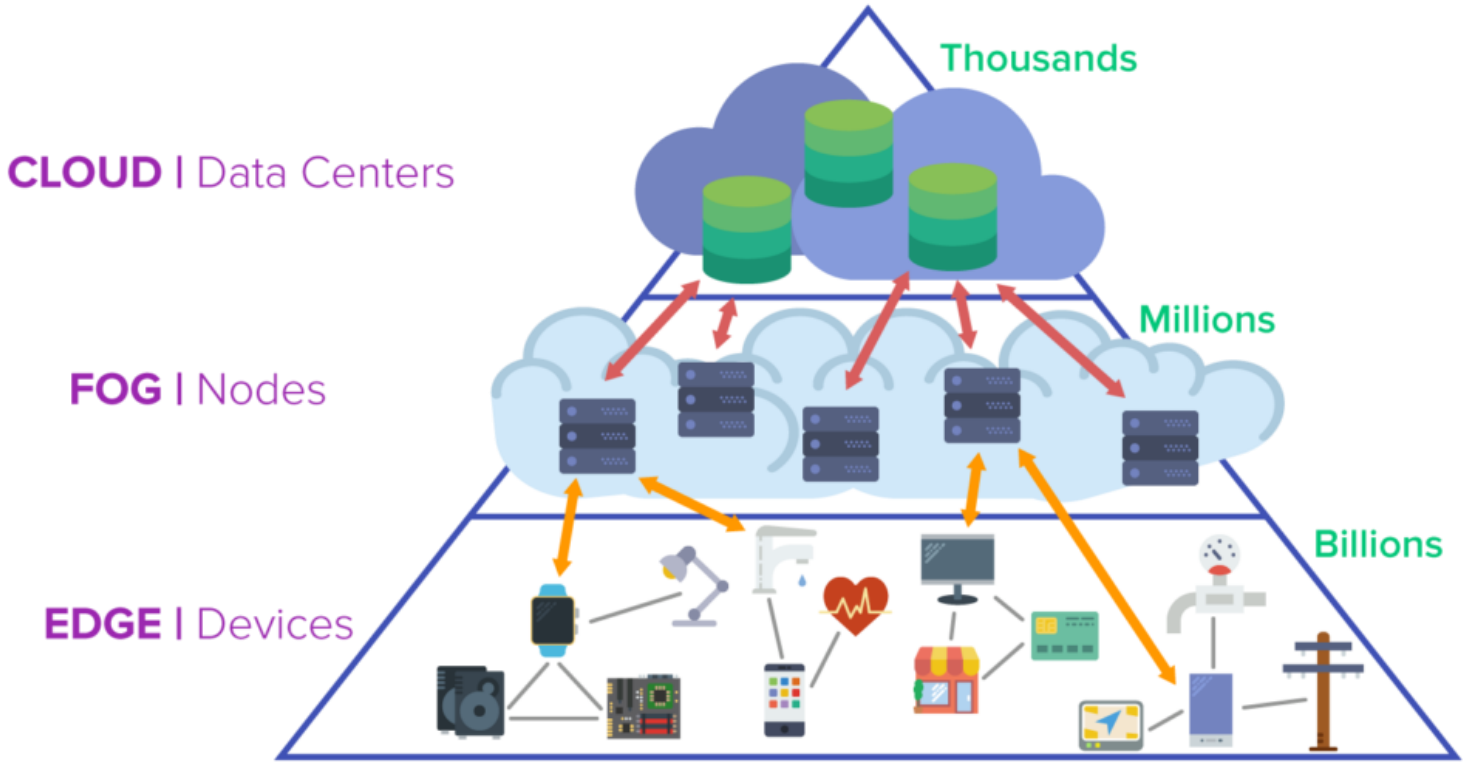

Fig. 1:I IoT system's layers.

Research Challenges

- Testing techniques and methodologies have long been developed and studied across software and hardware study areas.
- Due to the ***cross-domain particularities of the IoT***, long-pursued and pending research challenges from other study areas are now also becoming a problem of the IoT field.

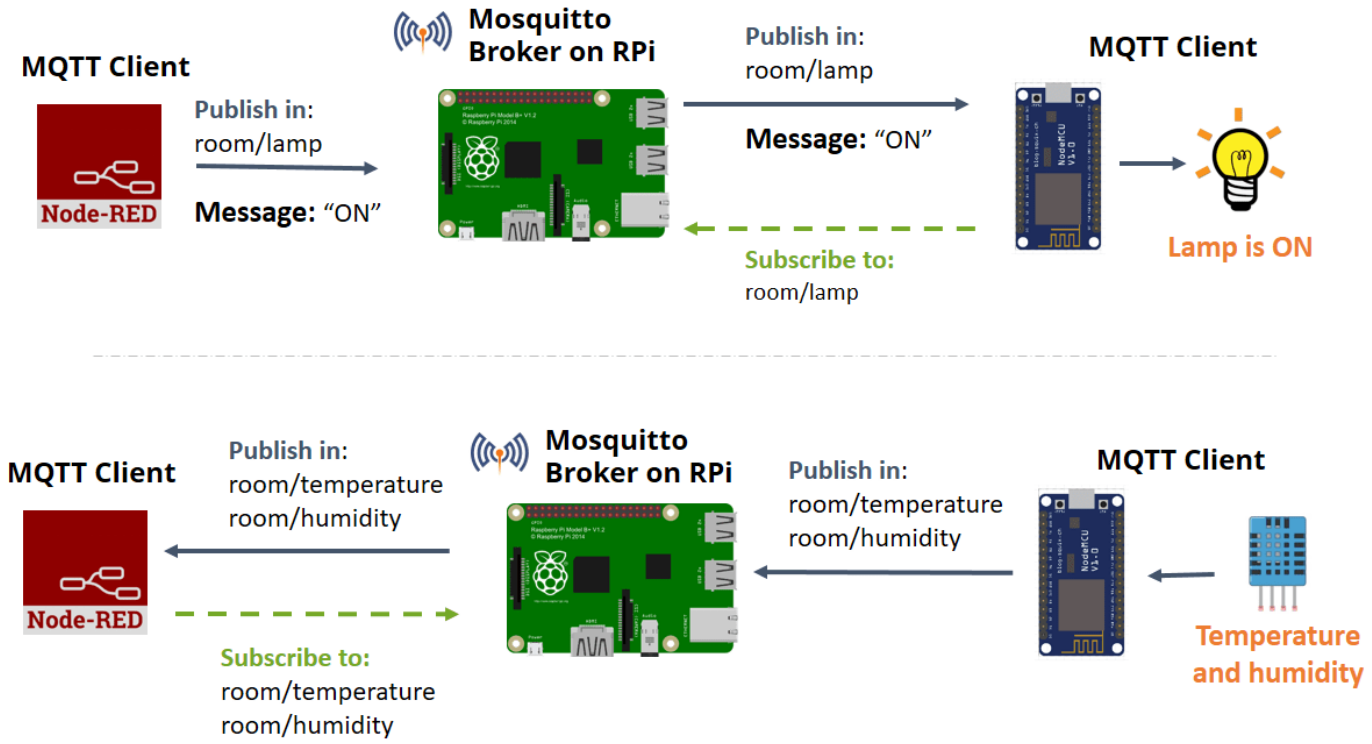

Fig. 2: Example scenario of the ***cross-domain particularities of the IoT (hw/sw)***.

Research Challenges

Heterogeneous Systems: Impact the integration and system-level testing. Although there are some techniques such as Manual Exploratory Testing, Combinatorial Testing and Search-Based Software Testing, there are still a considerable number of gaps.

Resulting in part from differences in industry focus and research focus.

Large-Scale Distributed Systems: Large-scale and highly-distributed systems lead to the appearance of new variables that need to be tested being some of them still open issues on the literature.

E.g.: Load testing and handling of dynamic behavior.

Research Challenges

Cloud-based Systems: Cloud computing has become ubiquitous nowadays, however there are still gaps on how to test cloud-based/cloud-connected systems.

E.g.: Design and test of elastic cloud-based solutions.

Embedded Software Systems: Devices typically have **constraints of memory and processing power**.

Also, these kind of devices are typically associated with **real-time needs and are prone to fail due to hardware problems** (e.g. power surge) which makes the **testing responses more volatile to environmental changes**.

IoT Testing Solutions

- A survey on the available tools for testing IoT systems was made, resulting in a total of 16 different tools/systems.
- An analysis of this tools and their documentation led to the definition of 10 *characterization* variables:

- Target IoT Layer (Edge, Fog, Cloud, Any)
- Test level (Unit, Integration, System, Acceptance, Any)
- Test Method (White-box, Black-box, Grey-box, Any)
- Testing Artifact (Code, Network, Application, Model)
- Supported Programming Languages (C/C++, Arduino, …)
- Test Environment (Simulator, Device, Platform, Physical Testbed)
- Test Runner (Local, Remote)
- Supported Platforms
- Scope/Target (Market, Academic)
- License (Close-source, Open-source)

Comparative Overview

Tool	IoT Layer	Test Level	Test Method	Testing Artifact	Prog. Lang.	Test Environment	Test Runner	Sup. Platforms	Scope	License
PlatformIO	Edge	Unit	White-box	Code	C/C++, Arduino	Device	Local , Remote	15+	Market	Closed
IoTIFY	All	Any	White-box	N/A	N/A	Simulator	Remote	N/A	Market	Closed
FIT IoT-LAB	All	Any	Any	N/A	N/A	Physical Testbed	Local, Remote	6+	Academic, Market	Open
ArduinoUnit	Edge	Unit	White-box	Code	Arduino	Device	Local	Arduino	Academic, Market	Open
MAMMotH	All	Integration, System	Any	Network	N/A	Emulator	Local	N/A	Academic	N/A
Cooja	Edge	Integration	Black-box	Network	C	Emulator	Local	Contiki OS	Academic, Market	N/A
TOSSIM	Edge	Integration	Any	Application, Network	Python, C++	Simulator	Local	TinyOS	Academic	Open
SWE Simulator	Edge	System	Black-box	Application, Network	XML, Visual	Simulator	Local	SWE Standard	Academic	N/A
SimIoT	Fog	Integration, System	Black-box	Any	N/A	Simulator	Local	N/A	Academic	N/A
iFogSim	Edge, Fog	Integration, System	Grey-box	Network	Java	Simulator	Local	N/A	Academic	Open
MobIoTSim	Fog, Cloud	Integration, System	Grey-box	Application, Network	N/A	Simulator	Local	N/A	Academic	Open
IOTSim	Cloud	Integration	Any	Application	N/A	Simulator	N/A	N/A	Academic	N/A
DPWSim	Fog, Cloud	Integration, System	Any	Application	WSDL	Simulator	Local	DPWS	Academic	N/A
SimpleIoTSimulator	Edge, Fog	Integration, System	Any	Network	N/A	Simulator	Local	N/A	Market	Closed
Atomiton IoT Simulator	All	Any	Grey-box	N/A	N/A	Simulator	Remote	N/A	Market	Closed
MBTAAS	All	Any	Black-box	Model	OCL	Platform	N/A	N/A	Academic	N/A

Comparative Overview

- A vast part of the available tools focus on a specific platform, language or standard.
- There is a lack of tools for testing certain artifacts such as:
  - *Security and privacy*
  - *Regulatory testing*
  - *Firmware/software upgrade (e.g. out-of-the-box continuous integration functionalities).*
- Most of the academic tools doesn't provide access to their source code or the software package.

Conclusion

The key features that differentiate IoT testing needs from the traditional systems are the **heterogeneous and large-scale objects and networks**.

These factors lead to an **increase on the complexity and difficulty** of testing IoT-based solutions.

There is a set of **old-known challenges** that are now having a direct impact on IoT systems.

- Further work needs to be done on the development of **testing solutions, automation procedures for testing and continuous integration** features.

We are still lagging behind on the best practices and lessons learned from the Software Engineering community in the past decades in what concerns to the IoT scenario.