CHALLENGES IN IOT TESTING

Ina Schieferdecker
TestNet, May 11, 2016
TALKING PLANTS, ANIMALS AND MORE

http://www.iot-a.eu/public
FURTHER FORECASTS

Connected Mobiles worldwide

Source: *Cisco Global Mobile Traffic Forecast Update, Gartner*

Global data streams in the Internet per Second in Terabyte

Source: *ITU ICT Facts and Figures 2015-2020*
IOT REFERENCE MODEL (ONE OF MANY)

Key Points
- IT–OT
- Decoupling
  Scalability
  Agility
- Interoperability
- Legacy Compatibility
- Analytics
- Integrated with the Enterprise

1. Physical Devices and Controllers (The “Things” in IoT)
2. Connectivity (Communication and Processing Units)
3. Edge Computing (Data Element Analysis and Transformation)
4. Data Accumulation (Storage)
5. Data Abstraction (Aggregation and Access)
6. Application (Reporting, Analytics, Control)
7. Collaboration and Processes (Involving People and Business Processes)
NEW ARCHITECTURAL PARADIGM

Hierarchical

Today

ERP
MES
SCADA
PLC
I/O

Upcoming

Orchestrated

Openness, Dynamicity, Scalability
CRITICALITY IMPLY HIGH QUALITY REQUIREMENTS

»Implementation of real-time enabled CPS solutions will place high demands on the availability of services and network infrastructure in terms of space, technical quality and reliability.«


Priorities of Quality, Time and Costs

ANYTHING NEW IN IOT TESTING ?!

Similar
- Protocol stacks
  - IETF-based: CoAP, MQTT, etc.
  - IEC-based: OPC-UA
  - ITU-based: M2M
- Application frameworks
  - Eclipse: Kura, Scada, etc.
  - Many others

Different
- Security
  - ISO: common criteria
  - Mitre: CWE list
  - Others
- Data
  - Semantic real-time data
- Protocol testing
  - Conformance
  - Interoperability
  - Performance
- Software testing
  - Component testing
  - Integration testing
  - System testing
- Security testing
  - Risk-oriented testing
  - Fuzz testing
  - Online testing
- Data quality
IoT solutions often are …

1. in harsh, unreliable environments

2. in highly dynamic configurations with large number of – typically diverse – sensors and actuators with open interfaces and

3. in resource-constrained environments

IoT test solutions need to …

- Integrate simulators for environmental conditions
- Systematically determine reference configurations
- Adjust and scale test configurations dynamically
- Be a real-time system by itself
- Support test scenarios for hybrid systems (both events and streams)

→ Test platform for the Internet of Things

© Fraunhofer FOKUS
INTEGRATION OF SEVERAL TESTING APPROACHES

Software Testing

System Testing

Protocol Testing

Security Testing

Test Automation

IoT Testing
TTCN-3 is the Testing and Test Control Notation
- Internationally standardized testing language for formally defining test scenarios. Designed purely for testing

```c
testcase Hello_Bob () {
    p.send("How do you do?");
    alt {
        []p.receive("Fine!");
        {setverdict( pass )};
        [else]
        {setverdict( inconc )}  //Bob asleep!
    }
}
```
DESIGN PRINCIPLES OF TTCN-3

- One test technology for different tests
  - Distributed, platform-independent testing
  - Integrated graphical test development, documentation and analysis
  - Adaptable, open test environment

- Areas of Testing
  - Regression testing
  - Conformance and functional testing
  - Interoperability and integration testing
  - Real-time, performance, load and stress testing
  - Security testing
TTCN-3 EXECUTION

testcase Hello_Bob () {
    p.send("How do you do?");
    alt {
        [p.receive("Fine!")
            {setverdict( pass )};
        [else]
            {setverdict( inconc )} //Bob asleep!
    }
}
A TTCN-3 TEST SYSTEM

ETSI ES 201 873-1   TTCN-3 Core Language (CL)
ETSI ES 201 873-5   TTCN-3 Runtime Interface (TRI)
ETSI ES 201 873-6   TTCN-3 Control Interfaces (TCI)
IMPLEMENTATION

ATS

Test System

SUT

Communication / Invocation
TTCN-3 DOMAINS: TELECOM

- Industrial use
  - Big companies with hundreds of TTCN-3 engineers: Ericsson, Nokia, Siemens, Motorola
  - Large distribution among SMEs

- Standardization bodies
  - Standardized test suites:
    - ETSI / 3GPP (LTE)/ OMA / TETRA and its members
  - IMS performance benchmarking:
    - Intel, HP, BT and others

- Test tool manufacturer:
  - Commercial Tektronix, Catapult, Nexus, R&S, Spirent, …
  - Free tools by Eclipse and academics

- Certification program based on TTCN-3: e.g. WiMax forum
Secure User Plane Location Protocol

Single MTC controls e.g.:
- UlpPort (Lup interface)
- IpcPort (IP configuration)
- smsPort used for SMS
- UtpPort for upper tester commands
- IpiPort (IP information, e.g. release)
- NwcPort: network bearer control, e.g. handover trigger
- SscPort: satellite simulation control, e.g. scenario trigger
- Cockpit systems
  - Edutainment
  - Head units
- Car-to-X communication
  - Car-to-car, car-to-roadside, car-to-backbone
  - Autonomic driving

**Telematics Applications in the Cockpit**
- Audio (CD / Radio), Video
- Telephone, SMS
- Navigation
- Speech recognition
- User interface for body electronic
- Interfaces to support distributed synchronous scheduling of components
- Interfaces to support transmission of continuous signals between components
- Concepts to deal with time and continuous signals
- Concept that allow advanced control flow for hybrid system testing
- Interfaces to support distributed synchronous scheduling of components
- Interfaces to support transmission of continuous signals between components
- Interfaces to support stimulation with and evaluation of continuous signals
- Interfaces to support access to time and sampling
**TTCN-3 EMBEDDED MODES**

**SIGNAL GENERATION BUILDING BLOCKS**

```plaintext
testcase signal_generation() runs on mtcType{
    seq{
        apply_noise(Throttle, 5.0, 5.0);
        apply_noise(Throttle, 10.0, 5.0);
        apply_ramp(Throttle, 10.0, 10.0, 2.0, 3);
        ...
    }
}
```
// accelerate vehicle until 35 ms and activate ACCS

cont{
    onentry{v_other.value := 25.0}
    phi_acc.value := 80.0;
}
until{
    [v_ego.value > 35.0] {
        phi_acc.value := 0.0;
        lever_pos.value := MIDDLE;
    }
}
// wait for several seconds
wait(now+10.0);
// evaluate
cont{
    assert(v_ego.value <= 38.0); }
until{
    [d_other.value < sd] { ...

1. Introduce a vehicle ahead
2. Accelerate the ego vehicle until its velocity rises to more than 35 m/s.
3. Activate the cruise control.
THE SIM\textsuperscript{TD} SET UP IN THE LAB
Elements

- Test control with TTworkbench and V2X framework (allow for synchronized stimulation and evaluation of V2X system interaction)
- Currently up to 4 IVS and 2 IRS systems to be connected to the test control over Ethernet
- Optional integration with ICT and other hardware possible
• IVS1: Generate situation
  WiperSystem := {
    Front := "normal",
    Rear := "idle" }

  WiperSystem := {
    Front := "fast",
    Rear := "idle" }

• IVS2: Check message reception
  – DENM message received ?

• IVS2: Check HMI interaction

EXAMPLE: WEATHER WARNING
DRIVE C2X REFERENCE TESTS

- Compatible with ETSI Standards
- Virtualized Test Environment

Tests available for:
- Stationary vehicle warning
- Road works warning
- Slow vehicle warning
- Traffic jam ahead warning
- In vehicle signage
- Emergency vehicle warning
- Emergency electronic brake lights

Example Traffic Jam Ahead Warning (TJAW):
Tests TJAW with different jam configurations by varying:
- number of vehicles in jam
- velocity of vehicles
- distance to EGO
- velocity of EGO

JAM is simulated by injecting CAM messages for the individual vehicles
SIM^{TD} REFERENCE TESTS

- 40 Communication tests and test variants
  - CAM variants
  - CAM frequencies, message life time handling etc.
  - DENM variants
- 20 Application tests
  - Testing event detection, propagation, handling and user notification for several V2X applications
- Reference circuit
  - Event handling and user notification for several V2X applications
- Reference circuit with load
  - Event handling and user notification for several V2X applications by applying networked and CPU load
- Goals: Integration, regression and acceptance testing

Project with Audi, Bosch, BMW, Continental, Daimler, Opel, Telekom, VW
Security testing solutions for six industrial domains

http://www.itea2-diamonds.org/

Ina Schieferdecker, Model Based Security Testing: Selected Considerations (Keynote) Sectest 2011, Workshop on the 4th IEEE International Conference on Software Testing, Verification and Validation Berlin, Germany
1. Fuzzing originally describes the random generation of test vectors (Miller et. al. in the early 1990s).

2. Fuzzing is about injecting invalid or random inputs in order
   • to reveal unexpected behaviour
   • to identify errors and expose potential vulnerabilities.

3. Ideally, fuzzers generate semi-valid input data, i.e. input data that is invalid only in small portions.

4. Depending on fuzzer’s knowledge about the protocol, fuzzers can generate totally invalid to semi-valid input data

→ Developed in DIAMONDS new behavior-fuzzing approaches

G&D Case Study
Banknote Processing Machines
G&D Case Study
Methodology

Risk analysis > Modeling > Test generation > Test code generation > Test execution

Attacker has access to the Message Router

Attacker sends "SQL injection" command messages over the Message Router

G&G_Test_Model_merge_FUZZED.uml

Test code generation

Test execution

Fraunhofer FOKUS
TRACING with CORAS, Papyrus, ProR and TTworkbench
EVALUATION AND OPTIMIZATION

Security test tool integration

Security test identification

Automated generation of test models

Fuzzing

Security test generation

Security passive testing/security monitoring

Test planning

Test identification

Test specification/modelling

Test generation

Test adaptation/realization

Test execution

Test incident reporting

Test selection/prioritization

Test selection/prioritization

Security risk assessment

Security test execution automation

Static security testing

How evolved is my security testing process?

How can I improve my security testing process?
CASE STUDY RESULTS

1. Collection of the experiences and results for all case studies
   - Case study experience sheets (DIAMONDS web site)
   - Case study experience report (ETSI document)

2. STIP Evaluation
   - Shows progress in all case studies
The RACOMAT Tool

Combines component based, low level risk assessment with security testing
- Risk analysis for component-based testing
- Reusable risk assessment artifacts
- Automated analysis of system components
- Integrates with external data bases like MITRE CAPEC and MITRE CWE

- Risk-Based Security Testing, Test-Based Risk Assessment and automation with the help of Security Test Patterns and Security Testing Metrics
- Semi-automated derivation of tests
- Automated execution of tests
- Test environments as part of test setups
- Combinations of real, virtualized and simulated components
- Integration of monitors and impairment components
- Management of test environments (configurations, versions, connections)
IPV6 TESTBED INFRASTRUCTURE

- Hybrid infrastructure running virtualized images and real physical devices
  - IPv6 Linux/FreeBSD/NetBSD/OpenBSD soft routers – XORP, Quagga, Zebra
  - Physical vendors’ hardware (e.g. Cisco Routers)
  - Virtualization and Virtualization Management - VMware ESXi, Virtual Box, Xen and OpenStack/CloudStack (in the pipeline)
  - Test automatization and reporting based on scripting and various tools (tcpdump, wireshark, pcap, Perl, Python, bash)
Experiments with IPv6 based Dynamic Routing (e.g. OSPFv3, BGP), QoS, and OpenFlow/SDN
FURTHER EXAMPLES

- HL7/IHE testing in eHealth
- TCMS testing in transport
- Performance testing in mobile communication
- Data platform testing in open data
- etc.
## Certified Tester for IoT ?!

<table>
<thead>
<tr>
<th>Main modules</th>
<th>Supplementary modules</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTEL</td>
<td>EL-ITP</td>
</tr>
<tr>
<td></td>
<td>EL-TM</td>
</tr>
<tr>
<td>CTA</td>
<td>AL-TM</td>
</tr>
<tr>
<td></td>
<td>AL-TA</td>
</tr>
<tr>
<td></td>
<td>AL-TTA</td>
</tr>
<tr>
<td>CTAL</td>
<td>Security</td>
</tr>
<tr>
<td></td>
<td>Test Autom</td>
</tr>
<tr>
<td></td>
<td>Industrial IoT ?!</td>
</tr>
<tr>
<td></td>
<td>Consumer IoT ?!</td>
</tr>
<tr>
<td>CTFL</td>
<td>Software Foundation</td>
</tr>
<tr>
<td></td>
<td>Embedded Systems</td>
</tr>
<tr>
<td></td>
<td>Foundation ?!</td>
</tr>
<tr>
<td></td>
<td>Agile</td>
</tr>
<tr>
<td></td>
<td>Automotive</td>
</tr>
<tr>
<td></td>
<td>MBT</td>
</tr>
<tr>
<td></td>
<td>Usability</td>
</tr>
<tr>
<td></td>
<td>Mobile</td>
</tr>
<tr>
<td></td>
<td>IoT ?!</td>
</tr>
</tbody>
</table>
is the network of excellence for the software development industry in German-speaking European countries.

1,400 globally active companies, specialists, institutions of higher education and research institutes are members of ASQF and share the commitment to guarantee quality standards in ICT.

is a leading provider of certification examinations all over the world, headquartered in Germany with subsidiaries in London, Boston and Amsterdam.

Focusing on IT professions, iSQI plays a large role in certifying the know-how of professionals in over 90 countries on 6 continents in 10 languages. In 2015, iSQI examines more than 22,000 individuals.
Working group on IoT Quality Engineering

- Challenges and risks
- Concepts, methods and approaches
- Syllabus, exams, training material
- Mainly members from industry (e.g. Festo, Siemens, SAP)

- If you are interested in status/results, please drop me an email: ina.schieferdecker@asqf.de
1. Combination of software, system, protocol and security testing

2. Need for high-degree of test automation

3. Management of distributed, flexible and/or virtualized test environments including test, simulation, SUT components and devices

4. Development of expertise and experiences in IoT Testing
CHALLENGES IN IOT TESTING