TTCN-3 in the Internet of Things(IoT), Testing in lossy environments

Anthony Baire, César Viho

UMR IRISA

cnrs Université de Rennes 1 INSA Rennes ENS Cachan Bretagne

June 2011

TTCN-3 User Conference

Summary

1. Context
2. IEEE 802.15.4 & 6LowPAN constraints
3. Addressing packet losses
    1. Hardware solutions
    2. Multiple runs
    3. Presenting the results in the TTCN-3 environment

Context of the work

- Focus on conformance & interoperability testing
- growing interest in testing in unreliable environments
- IPv6 Ready Logo Committee
  - Certification program for the IPv6 protocol suite

- Partnership with the IPSO Alliance *(IP for Smart Object)*
  - Design tests for IPv6 embedded devices
  - Targets 6LoWPAN, -ND (neighbour discovery), -HC (header compression) and RPL (routing) protocols

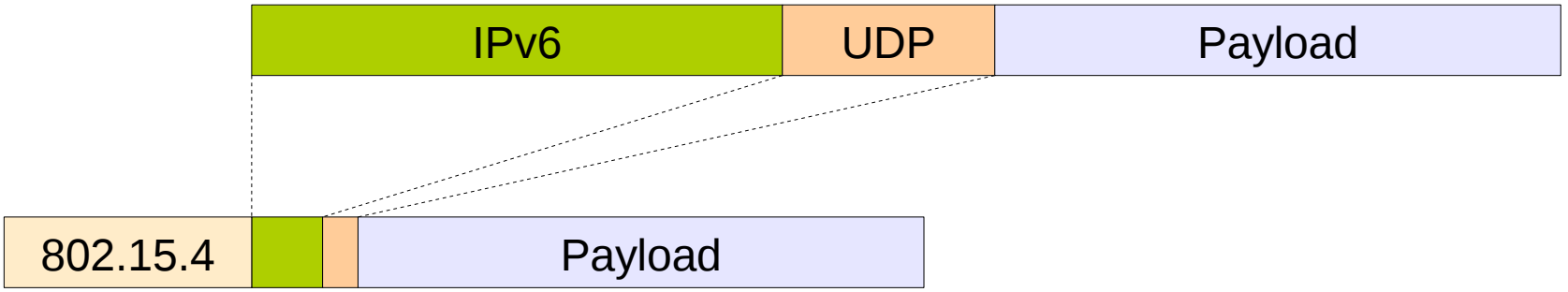
IEEE 802.15.4 (typical 6LowPAN link layer)

- Low power radio link-layer
  - Low power (~1mW)
  - Lossy (using the 2.4Ghz band)
  - Low rate (20kbps to 250kbps)
  - Tiny frames (127 bytes)

- Applications
  - sensor networks
  - personal area network (home automation, ...)

IETF 6LowPAN overview

- IPv6 adaptation for Low power wireless networks (especially IEEE 802.15.4)
  - features
    - header compression, fragmentation (tiny frames)
    - mesh routing
    - support for sleeping nodes



Link reliability vs. test requirements

- Link layer not reliable
- Packet losses
- Issues for testing
  - Observability issues
    → risks for
      - False positive (permissiveness)
      - False negative (bias)

Possible solutions

- hardware solutions
- run the tests multiple times
- design the testcases to produce an 'inconc' verdict in case of (suspected) packet loss
- monitor the environment during the tests

Hardware solutions

- Possible strategies
  - Bypass the lossy medium
    - eg. connect the implementation directly w/ a coaxial cable

      → generally not possible
      (embedded devices, embedded antenna)

  - Minimise the probability of packet loss
    - difficult: same band as wifi signals (20db stronger)
    - would need a Faraday cage
    - how about other physical media ?

Multiple runs approach

- Run each test case multiple times

  → increases the chances of having a clean run

- Issue: how to distinguish biased verdicts from correct verdicts?
  - manual case-per-case analysis too cumbersome
  - need a way to prioritise these verdicts

use verdicts precedence

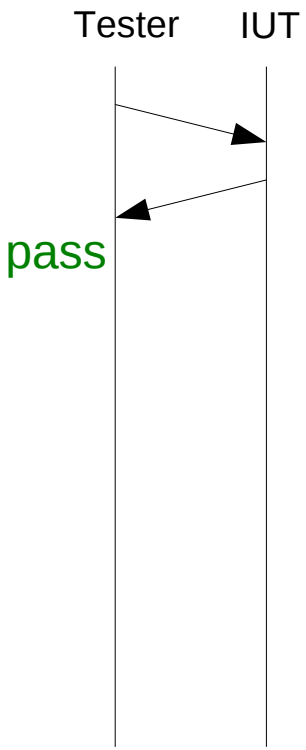
- Possible Solution: return “inconc” in case of suspected packet loss

  (i.e. consider packet losses as a property of the SUT)

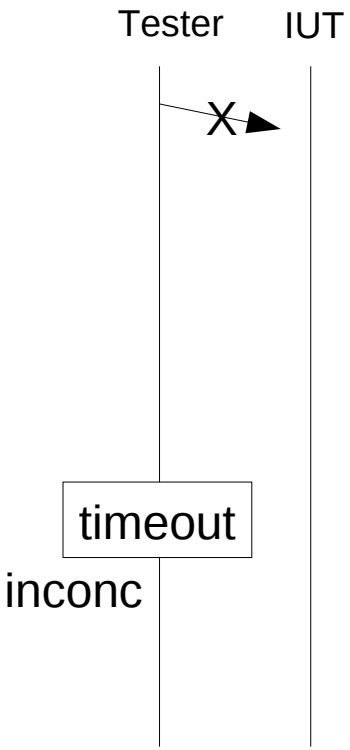
→ then combine the verdicts from multiple runs

Example 1: reply expected

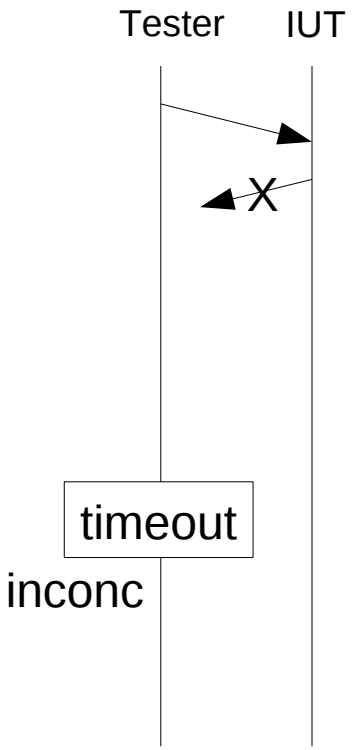
Normal case



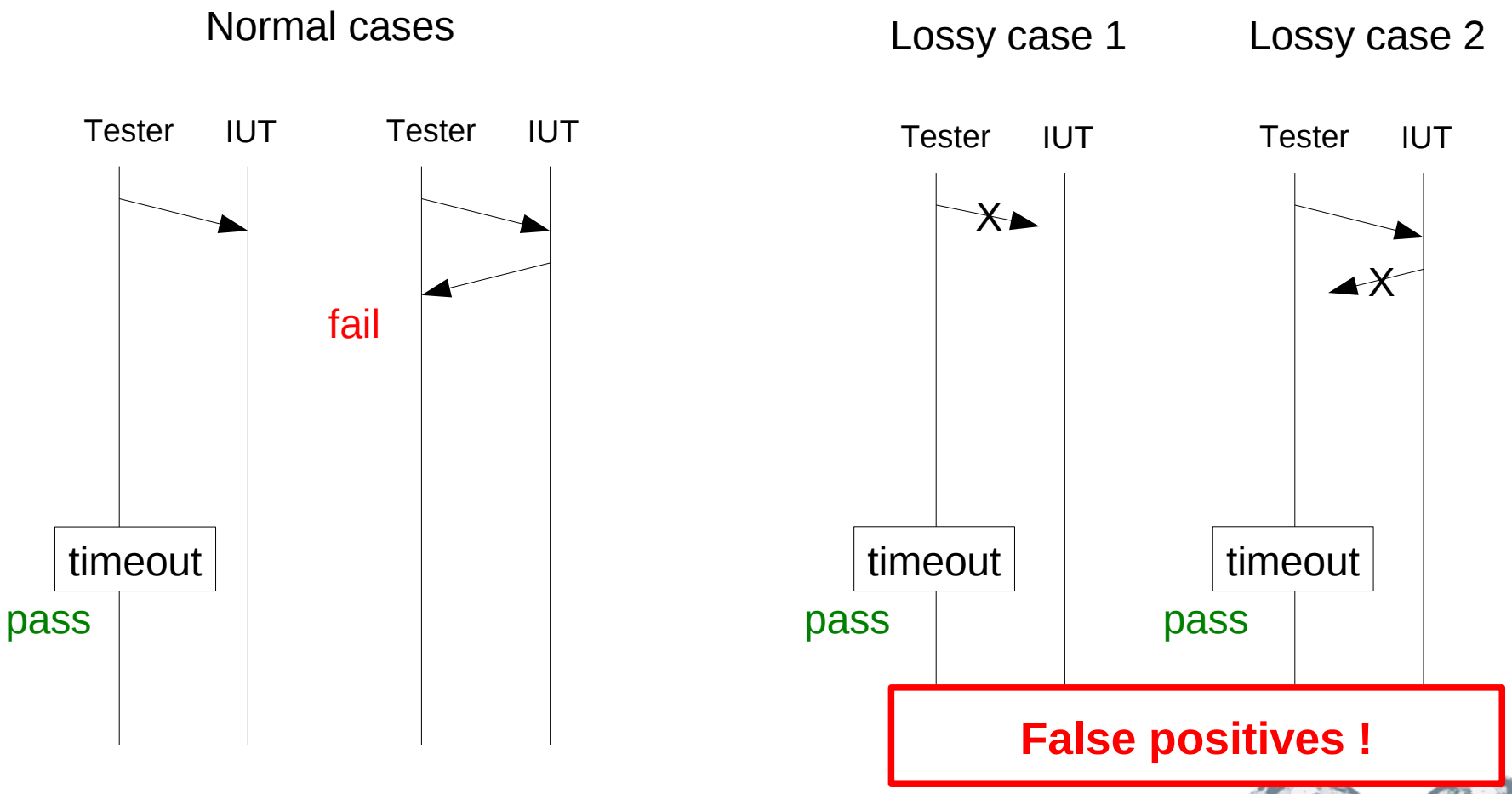
Lossy case 1



Lossy case 2

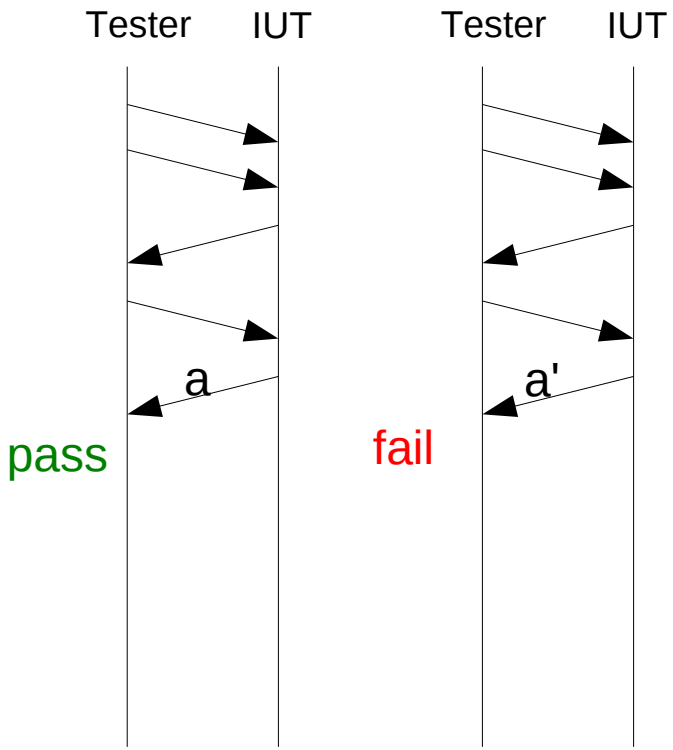


Example 2: reply not expected

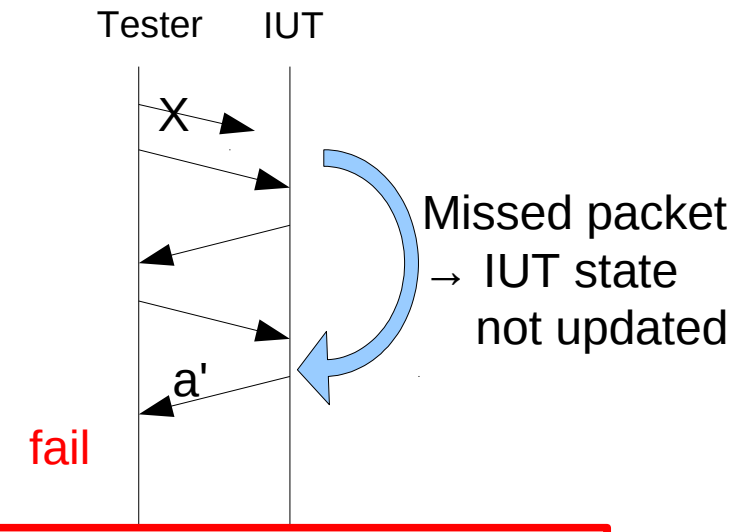


Example 3: stateful behaviours

Normal cases



Lossy case



False negative !

Not easily managable in practice

Monitor the environment

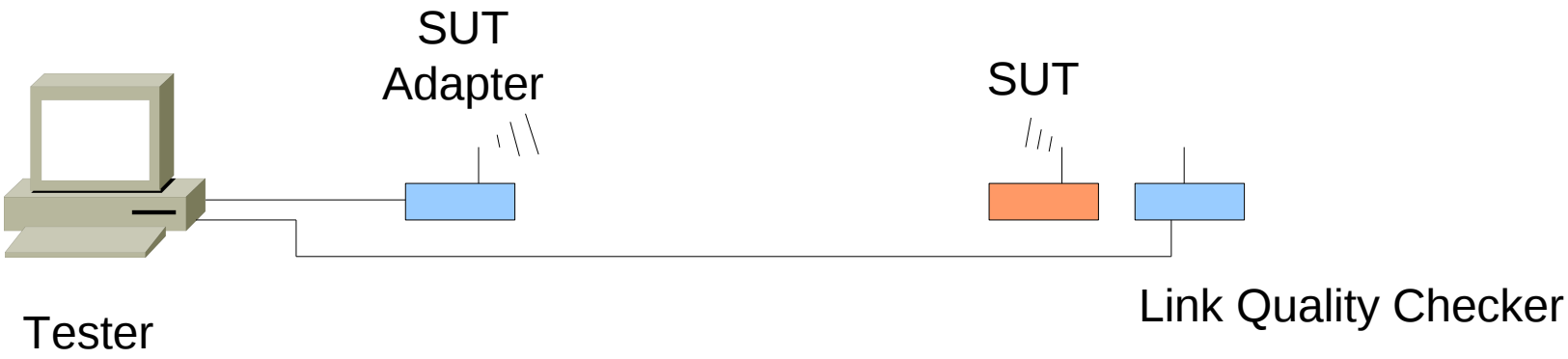
- monitor link quality during the test execution (background noise)
- objective: evaluate a level of confidence of the verdict of each test run

Using a second transceiver

- Use a second transceiver to check if frames sent to the SUT are received with good quality
- For each received frame, the transceiver returns the quality of the signal

→ higher signal quality means higher confidence in the test verdict

Using a second transceiver



→ detection of forward losses (Tester → SUT)

Issue
→ how to detect losses from SUT the tester?

Presentation of the results & TTCN-3

- Link quality monitoring feasible in the System Adapter. How to report it and re-execute the testcase ?
  - directly in triEndTestCase() → return TRI_ERROR when the actual verdict is not reliable
  - interact with the testcase (external function/port) to report the level of confidence

Issues & possible evolutions in the TTCN-3 environment

- Issue :
  - in case of multiple runs, all instances of the same testcase are equally presented to the user
  - some post-processing is required to indicate which one is relevant

    → it would be useful to have a way to highlight (or select) which testcases instances to present in the log summary

Conclusion

- Lossy medium induces uncertainty in the test
- Some solution identified:
  - bypass the lossy medium (hardware solution)
  - multiple runs
  - assumes packet loss is a property of the SUT (inconc) (use the 'inconc' verdict)
  - monitor the environment (level of confidence)
- Need a way to highlight the relevant testcases instances in the test results

Questions ?

Contacts: abaire@irisa.fr / viho@irisa.fr