Testing Strategies for IoT Systems: A Systematic Literature Review

by Goian Sergiu
BBU Cluj-Napoca

Motivation

IoT = rapid growth in mainstream industries (healthcare, manufacturing, transportation etc.)

resource constraints (limited CPU / RAM / Storage / bandwidth) decrease reliability

Devices heterogeneity makes conventional testing approaches insufficient or even impossible

Failures in these systems can lead to service disruptions or security breaches, highlighting the importance of specialized testing frameworks

Research Gaps: while numerous studies exist, there is a lack of consolidated insights on which testing methodologies are most efficient and where automation falls short

Research Questions

RQ1: Most prevalent testing methodologies for security, reliability and performance?

RQ2: Main challenges preventing full test automation in integrated environments?

Systematic Review Methodology

Kitchenham's SLR guidelines

Allows a rigorous and reproducible synthesis of existing literature, helping identify current trends, gaps, and best practices in IoT testing

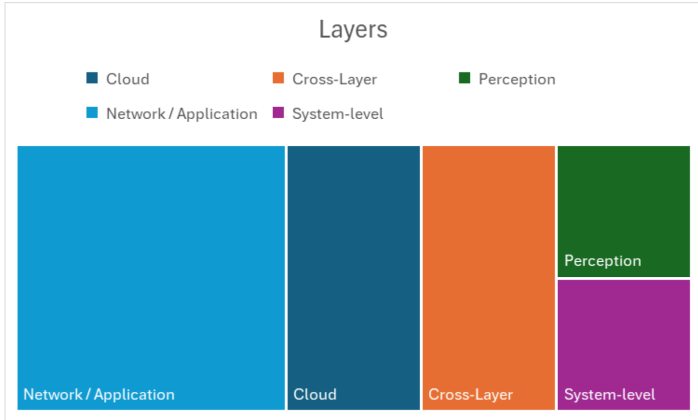
Search strategy: Libraries (IEEE, MDPI, ResearchGate) filtered by keywords that are meant to capture literature at the intersection of IoT and software quality (ex: "IoT Testing" and "Quality")

Selection: from 30 previous candidate articles => **only 10 primary studies** from 2015 to 2025 via snowballing filtered based on their relevance to IoT testing methodologies and accessability

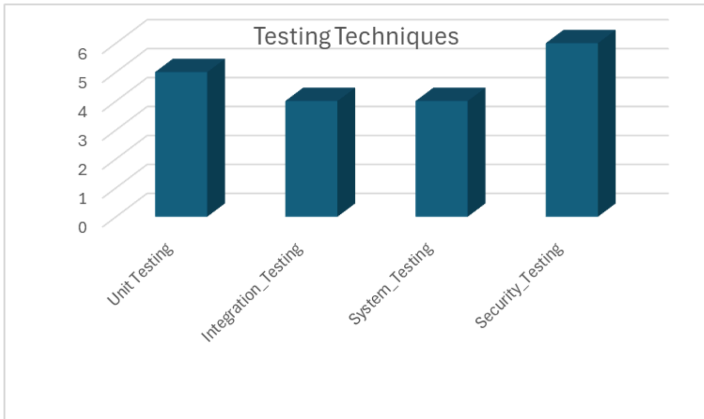
Analysis & Core Findings

The IoT Testing Landscape

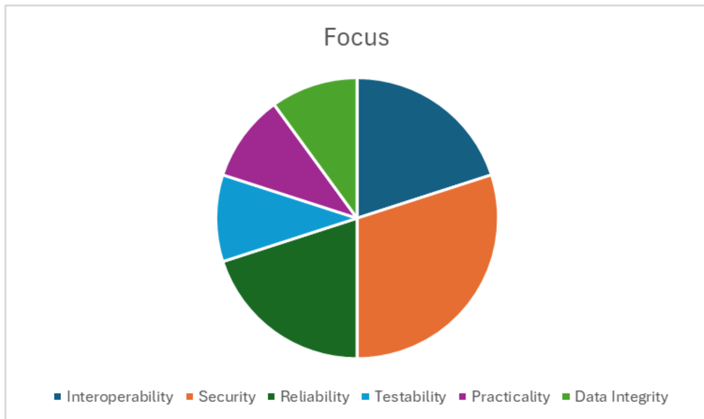
Layer Distribution



Prevalent Testing Techniques



Focus



Synthesis of RQ1: Prevalent Methodologies

Analysis shows that unit-level and security testing are the most prevalent approaches in IoT validation

Literature shows a transition from black box to Fuzzing and LSTM / RNN intrusion detection (with 98% accuracy)

Use of Model-Based Testing (MBT) for automated integration, for instance PatrIoT framework

Synthesis of RQ2: Barriers to Automation

Proprietary and asynchronous protocols break "state-unaware" automated tools

High cost of physical testbeds leads to over-reliance on emulators that fail to replicate real world latency and packet loss

Despite academic focus on automation, industry still relies on manual system-level integration

Challenges & Future Outlook

Key Takeaways (Results)

Specialized IoT audits uncover **40% more vulnerabilities** than generic network security audits

MBT significantly reduces manual effort in continuous integration (CI) pipelines

Trade-offs

!!! up to 20% latency increase when using Blockchain for data integrity, a critical finding for real-time systems, while ML require high computational cost

Conclusion & Future Work

Need for metrics concerning device-level fault tolerance

Developing "state-aware" fuzzing and lightweight ML models for the perception layer

Experiment: light ML deployed on constrained sensor nodes