

WHITE PAPER

IoT Testing **Challenges** & Approaches

Businesses adopting Internet of Things (IoT) need to deliver robust, high-quality solutions. Testing complex IoT solutions with large number of devices, which continuously generate data, poses challenges for internal test teams.

Independent IoT Testing Services providers can ensure quality of complex IoT solutions with a multidisciplinary approach, well-planned test strategies, and use of appropriate automation, simulation, virtualization and measurement tools.

Introduction

IoT Testing Challenges

IoT Testing Approaches

Types of Tests

Test Tools & Frameworks

Conclusion

Introduction

Internet of Things (IoT) is impacting and significantly transforming products and services businesses across industries worldwide. Gartner® Research says that there will be approximately 20 billion connected “things” by 2020.

Delivering robust, high-quality IoT solutions quickly to the market is a key requirement for businesses. Testing large numbers of devices that are often heterogeneous, and which are continuously generating data, poses significant challenges in terms of scale, velocity and variety for internal test teams. Further, traditional software application quality control approaches are inadequate for such IoT solutions.

Ensuring quality of such complex IoT solutions requires a multidisciplinary testing approach. Besides coming up with a well-planned test strategy, the use of appropriate automation, simulation, virtualization and measurement tools is vital.

This white paper shares Thinxstream's experience on the key testing challenges, testing approaches, and testing solutions for achieving high-quality IoT solutions.

IoT Testing Challenges

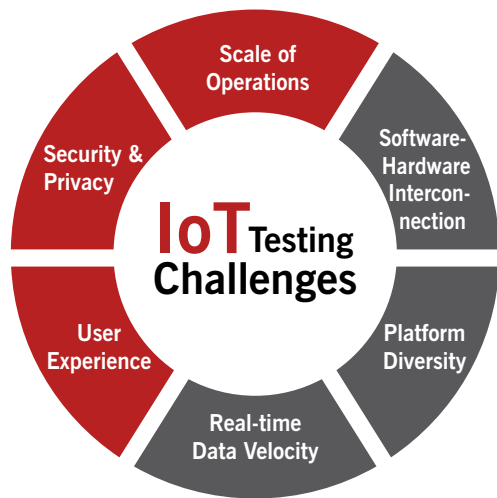


Figure 1. IoT Testing Challenges

IoT solutions are complex from multiple perspectives. Firstly, the diversity of solution components involved – device hardware, application software, server software, network and, client platforms. Secondly, the massive scale and throughput at which they are expected to function across networks. Thirdly, the myriad user and environmental situations under which they are expected to operate.

Scale of Operations: IoT solution deployment entails thousands of interconnected devices, which connect to servers (on-premises or in the cloud) over near real-time networks. Server infrastructure is built on multiple interconnected services and applications from different vendors. Testing such a complex, multivendor environment and simulating real-time situations is always a challenge.

Software-Hardware Interconnection: Testing an IoT solution is not limited to the application or the hardware. It requires an integrated IoT testing approach for this interconnected and dynamic environment. Apart from routine functional and non-functional testing of individual software and hardware components, it is important to test practical scenarios that consider the interactions between them.

Platform Diversity: In such a diverse field, there are many software, firmware and hardware platform variants. In addition, there are different network protocols and mechanisms for device-to-server connection such as MQTT, HTTP, COAP and WebSockets. Testing for all possible combinations is not practical. Shortlisting relevant test scenarios requires good understanding of end-use situations, domain knowledge, and a platform-agnostic and automated test suite.

Real-time Data Velocity: Thousands of connected devices reporting telemetry data periodically places a significant load on the network. Challenges from unreliable network hardware and Internet connections could impact device performance and ultimately the IoT solution. Since these devices are mostly remotely connected, such situations lead to frustrated end-user

experiences. Testing responsiveness of devices and applications for all such real-life situations is a constant requirement throughout the IoT solution development lifecycle.

User Experience: For any IoT solution, seamless and consistent user-experience across mobile (typically iOS®, Android™) and desktop (typically Windows®, Mac®) environments is key. Further, preserving native experience on mobile platforms is also an implicit requirement. Testing needs to consider these diverse user environments across multiple brands, versions and screen sizes.

Security and Privacy: Networked devices and applications exposed on public Internet are always vulnerable to being hacked. Conforming device and applications against the prescribed security standards is vital. As IoT grows, hackers are constantly trying to find system weaknesses. Constant security upgrades and testing is a must in today's environment.

IoT Testing Approaches

Ensuring quality of such complex IoT solutions requires a multipronged testing approach. It requires a well-planned IoT testing strategy that is comprehensive and is constantly evolving with changes. It includes test management tools, test classes, test lab setup comprising simulators, ready-made tools, and extensible frameworks. Tests address individual components of the IoT solution including hardware and software, as well as the integrated solution.

Types of Tests

Usability Testing: A usability test ensures that the interface of the device and the application meets user's expectations. The primary focus of these tests are to ensure ease of use for most basic operations, responsiveness, preserving nativity, graceful handling of errors and ability to use device/app without training or a guide.

Security Testing: Security in its most basic form means that authorized access is granted to the protected device and its data and unauthorized access is restricted. Testing is done using threat

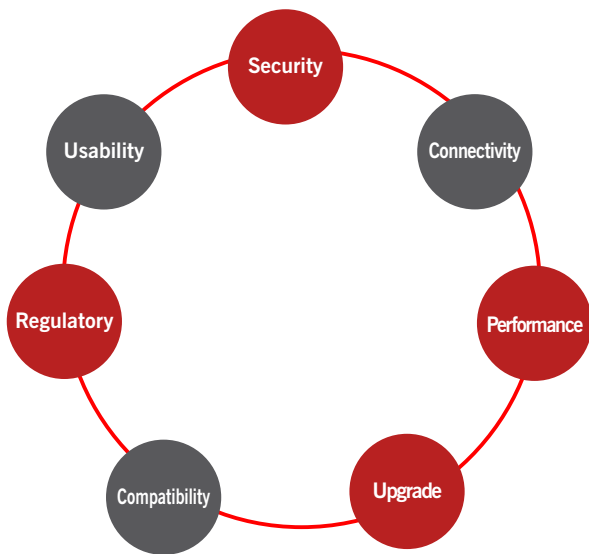


Figure 2. Types of Tests

modeling tools, static code analysis tools and runtime check tools, which subject the device and application to a range of simulated threats. Security tests also encompass checks for OWASP Top 10 threats.

Connectivity Testing: This testing involves checking the device and application behavior on subjecting the network through a load, intermittent failures, and total loss of connectivity. By inducing these real-life scenarios, the robustness of the device, edge, platform and application is checked.

Performance Testing: On the device, these tests check their responsiveness to user actions and on a platform layer, it checks ability to handle spikes in traffic gracefully. They are based on metrics for assessing the responsiveness of the device/application and underlying system performance. Load generators and

performance measuring tools on the cloud rate system performance under normal and full load.

Compatibility Testing: In a complex IoT environment, devices, network, platforms, applications and end-user desktops/mobiles work in tandem. Each one of them has a high degree of variability in terms of the firmware and hardware models and versions; network type, speed, protocols and versions; operating system type and versions; browser type and versions; screen sizes and display resolutions to name a few. It's important to test the application in all possible combinations of these versions to reduce failures in the field.

Compliance & Certification Testing: A well tested IoT product may also require the right certifications to enter the market. IoT devices generally have to meet various certification requirements for the network, protocol compliance, device drivers, app store submissions, etc.

Upgrade Testing: Whenever firmware, software or hardware updates or upgrades occur, it calls for thorough regression testing as failures may arise due to compatibility issues. Post upgrade, data preservation and a smooth system restart are critical. To address this, special tests are often

performed in a staging environment before upgrades are pushed OTA to devices and on server systems.

Test Tools and Frameworks

In order to execute the wide range of IoT testing listed above in a staging environment, use of right automation, simulation, virtualization and measurement tools is quite vital. Listed below are some of the tools that could be used in different situations:

Device/Protocol Simulators: Devices which are standards compliant can be often simulated using tools. They can be simulated in large numbers as well as configured to simulate the required real-life states.

Record and Play Tools: Be it devices or applications, system and user data/actions can be recorded and replayed on simulators and apps as a means of automating the test execution.

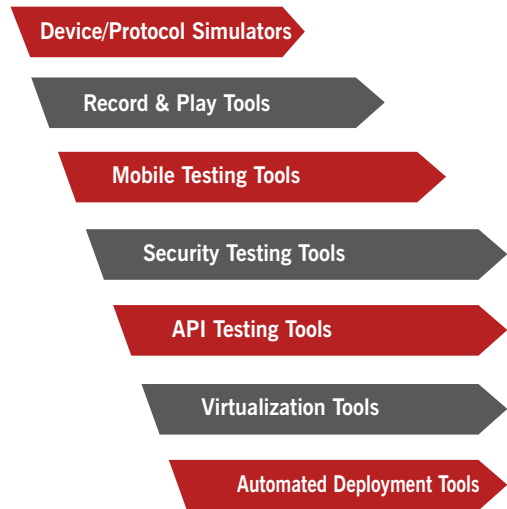


Figure 3. Test Tools and Frameworks

Mobile Testing Tools: They provide automated functional mobile testing that replicates end-user experience and ensures that the application works as expected.

Security Testing Tools: They can be classified into threat modeling, static code analysis and runtime threat inducing tools. Tools such as Micro Focus® Fortify on Demand, OWASP ZAP, VCG and Microsoft® Threat Modelling Tool identify threats, prioritize them and also provide recommendations on how to fix them.

API Testing Tools: Increasingly solutions are now built using REST APIs and Web services. Tools such as Postman, SoapUI, Progress® Telerik® Fiddler™, etc. test their connectivity, response, and performance.

Virtualization Tools: They enable economic and timely execution of

compatibility tests without the requirement for investment in different hardware, operating systems, browsers, databases, platform services, etc.

Automated Deployment Tools: They are used to programmatically create virtual machines either on-premise or in the cloud, rapidly commission managed services and configure and deploy custom-built services and applications. Tools such as Foreman, Ansible Tower® and Katello ensure the building of staging setup so that automated and manual tests can be triggered on time in continuous build, integration and deployment environments.

Other Tools: Additionally, test case and defect management tools and proprietary tools can improve productivity, speed, and effectiveness of quality control execution.

Conclusion

IoT solutions are complex and challenging given the multiple components and interactions between them. Varied IoT tests can ensure a quality IoT solution. However, executing them requires a good testing strategy with the use of appropriate test tools.

As an IoT testing services provider, Thinxstream is capable of ensuring quality IoT solutions. Thinxstream has expertise in an array of tools, processes, and best practices for managing test scope and schedules, test scenarios, and test data. It is well-versed in challenges that emanate from scale, diversity and remote access to such complex environments. By leveraging the IoT expertise built over a decade, Thinxstream ensures cost-effective, quality and timely delivery of IoT testing services.

Thinxtream Technologies is a global software company with a portfolio of innovative software platforms, products, components, solutions, patents, competences and services for Internet of Things (IoT) across several industry verticals and applications, successfully enabling leading customers, including Fortune 500 companies, meet their application, product and business goals.

Interested in learning more? For more information contact:

Thinxtream Technologies Pte. Ltd.

220 Orchard Road #05-01
Midpoint Orchard
SINGAPORE 238852

Phone: +65 66358625

Email: info@thinxtream.com

 www.thinxtream.com

Thinxtream Technologies, Inc.

10260 SW Greenburg Road
Suite 400 Portland, OR 97223,
U.S.A

Phone: +1 503 293-3598

Email: info@thinxtream.com

 [LinkedIn/thinxtream](https://www.linkedin.com/company/thinxtream)

Copyright© 2018, Thinxtream Technologies Pte. Ltd. All Rights Reserved. The information in this publication supersedes that in all previously published material. For the most up-to-date information, please visit our website at www.thinxtream.com.

Thinxtream is a registered trademark of Thinxtream Technologies Pte. Ltd. Mac is a registered trademark of Apple, Inc. Android is a trademark of Google LLC. iOS is a registered trademark of Cisco Systems, Inc. Microsoft, Windows are registered trademarks of Microsoft Corp. Gartner is a registered trademark of Gartner, Inc. Micro Focus is a registered trademark of Micro Focus (IP) Limited. Progress, Telerik are registered trademarks and Fiddler is a trademark of Progress Software Corporation. Ansible Tower is a registered trademark of Ansible, Inc. All other trademarks are the property of their respective owners.

All prices, specifications and characteristics set forth in this publication are subject to change without notice.

TT-WP-002-1-0718

