

CUSTOMER CASE STUDY

How a Telecom Provider Found the Network Defects Emulators Could Not Reproduce

Powered by the Private Real Device Cloud from TestMu AI

SUCCESS METRICS

100%

Of mobile testing was manual before the change

3 in 4

Network defects found on live SIMs did not reproduce on emulators

1 quarter

To first automated regression pass across priority mobile journeys

4 product lines

Wireless, wireline, fixed wireless access, and 5G standalone network slicing in scope

INDUSTRY

Telecommunications

LOCATION

Global operations, business products division



Telecommunications

Table of Contents

Overview	03
Company Snapshot	03
When the Network Is the Product	04
What the Team Was Looking For	05
Testing on the Network, Not a Simulation of It	06
What Changed: Defects, Throughput, and Participation	07
Beyond Mobile: Fixed Wireless and Network Slicing	08
About TestMu AI	09

Overview

This case study is an illustrative composite built from real, recurring customer pain points, anonymized for confidentiality.

The organization is a large telecommunications provider whose business products division sells wireless, wireline, fixed wireless access, and 5G standalone network slicing across four customer segments: small business, mid-market, global enterprise, and public sector.

Within that division sits a testing center of excellence. The team performs final-stage testing in the production environment, after development teams have completed their own testing, and currently concentrates on the division's largest and most significant products while expanding its remit.

Three surfaces fall inside its scope: customer-facing portals used for analytics and self-service, the ordering systems used by sales representatives, and the division's mobile applications.

That scope held up under a manual process while the estate was smaller. It stopped holding up as the product range grew.

"An emulator can tell us the screen rendered. It cannot tell us what happens to a session on a live network."

– QA Leadership, Telecommunications Provider

Company Snapshot

INDUSTRY

Telecommunications

DIVISION

Business products, serving small business, mid-market, global enterprise, and public sector

PRODUCT LINES

Wireless, wireline, fixed wireless access, 5G standalone network slicing

SURFACES TESTED

Customer portals, sales-ordering systems, mobile applications

QA STRUCTURE

Testing center of excellence, final-stage testing in production

TESTMU AI PRODUCTS USED

Private Real Device Cloud, Real Device Cloud, KaneAI

When the Network Is the Product

Testing across the division's estate was entirely manual.

The scope is the difficulty. A wireless product and a fixed wireless access product do not behave the same way, and a 5G standalone network slicing product introduces network behavior that cannot be observed from a browser session on a desktop machine. The same portal has to work for a small business customer and for a global enterprise administrator. Each combination is a separate thing to verify, and each was being verified by a person.

Under the team's existing approach, every new product, market, and device added another pass through the same sequence.

THE EXISTING PATH



Scaling that model means either adding people in proportion to products, or finding tooling. It also left the network itself largely untested, because a desktop browser session and an emulator cannot reproduce what a handset does on a live cellular connection.

The constraints the team was working against

- Manual verification did not scale with a product range spanning wireless, wireline, fixed wireless access, and 5G standalone network slicing.
- Network behavior, which is the product for a carrier, could not be verified from emulators or simulated connections.
- Testing runs in the production environment and against internal systems, which constrains where that testing can take place.
- Any tooling adopted had to be usable by the wider product community, not only by test specialists.

What the Team Was Looking For

When the team evaluated how to change its approach, the requirements were specific to a carrier.

Physical devices carrying live SIMs

Mobile applications belonging to a network operator have to be verified on the network, not on a simulation of it.

A dedicated environment

Testing runs in production, against internal systems, for a division whose customers include public sector accounts.

Usable beyond the QA function

The wider product community needed to work in the same environment without a test automation background.

Findings in existing systems

Defects and test cases had to land in the tools the division already runs its release process on.

TestMu AI was already under evaluation for real device workloads. Private devices extended that into the part of the process the team could not reach at all, which was the network itself.

Testing on the Network, Not a Simulation of It

For a telecommunications provider, the network is the product. That distinction determined the approach.

The team adopted the TestMu AI private real device cloud, with physical SIM cards installed on dedicated devices, for mobile application testing across its wireless and fixed wireless access products.

Three capabilities decided private devices over shared ones. A private device can hold a physical SIM, which makes carrier network testing possible on the network itself. It can be configured to match the conditions the team needs to reproduce, and it holds that configuration between sessions. And a dedicated environment keeps production and internal-system testing isolated.

Real devices across regions

Physical devices available in multiple regions, supporting testing against the markets the division sells into, with IP geolocation testing across more than 60 countries for its global enterprise customers.

Secure connectivity to internal systems

VPN and tunnel support, which the team requires because its final-stage testing runs against production and internal tooling rather than an isolated staging environment.

Network conditions on both sides

Live SIMs cover real carrier network behavior. Network throttling covers degraded conditions on demand. Together they let the team verify how the application behaves when connectivity is poor as well as when it changes.

Authoring the wider team can use

Tests are described in natural language rather than written as framework-specific code, which gave the division a route from an entirely manual practice toward automation without requiring every participant to script. Self-healing maintenance, visual validation, and API testing sit alongside it.

Findings route into Jira and qTest, so evidence lands in the systems the division already uses to manage defects and test cases.

What changed: defects, throughput, and participation

The first measurable change was where defects were found. Testing on physical handsets with live SIMs surfaced a class of defect the previous approach could not reach. Of the network-related defects identified in the first quarter, roughly three in four did not reproduce on emulators or over simulated network connections.

These were session and connectivity behaviors: sessions dropping as a device moved between network conditions, authentication state lost on a cellular connection but retained on Wi-Fi, and application behavior that varied by network configuration. On a wireless product, these are not edge cases. They are the conditions most customers use the product in.

The second change was throughput. The center of excellence moved from entirely manual testing to an automated regression pass across its priority mobile journeys within a quarter, running on the same dedicated devices used for exploratory work. Manual effort shifted toward the exploratory and final-stage verification the team is best placed to do.

The third change was participation. Because tests are authored in natural language and run in interactive sessions, people outside the QA function were able to use the same environment. That was a stated requirement at the outset, and it widened the group able to verify a product release beyond the test specialists.

Before



Entirely manual verification

Every product, market, and device combination verified by a person, with network behavior largely untested.

Quarter one



Defects found on live networks

Three in four network defects identified did not reproduce on emulators or simulated connections.

By quarter end



First automated regression pass

Priority mobile journeys covered automatically, on the same dedicated devices used for exploratory work.

"We were verifying that the application worked. We were not verifying that it worked on our own network. Those turned out to be different questions."

– QA Leadership, Telecommunications Provider

Beyond Mobile: Fixed Wireless and Network Slicing

The first quarter settled a question the team had been carrying, which was whether network conditions could be brought inside the test cycle rather than discovered after release. Having established that they could, the division is extending live SIM coverage to the products where network behavior is the characteristic under test.

Fixed wireless access and 5G standalone network slicing are the immediate targets. Both are products whose value is defined by how the network performs, and neither can be meaningfully validated through an interface alone. A slicing product that renders correctly but does not hold its allocated network characteristics has not been tested. It has only been looked at.

In parallel, the team is scaling automated coverage across the remaining surfaces in its scope: the customer-facing portals used for analytics and self-service, and the ordering systems used by sales representatives. Both carry the same requirement that drove the original decision, which is that the people who understand the product should be able to verify it without depending on scripting capacity.

For a division selling across small business, mid-market, global enterprise, and public sector customers, that combination determines how quickly a product change can reach all four.

About TestMu AI

TestMu AI (Formerly LambdaTest) is a fully autonomous agentic quality engineering platform that empowers teams to test intelligently, smarter, and ship faster. Over 10,000+ customers and 2 million+ users across 132+ countries rely on TestMu AI for their testing needs.



1.2 Bn+
Tests



2M+
Users



10K+
Enterprises



132+
Countries



Exploratory Testing

Enhance web and app quality to ensure seamless user experience with real-time, live, exploratory testing on 10,000+ devices.



KaneAI

Boost testing efficiency with an AI platform that uses natural language to create, debug and evolve tests.



Test Manager

Streamline test creation, management, & reporting for improved efficiency with AI - native unified Test Manager.



Automation Cloud

Accelerate product releases with secure, scalable, end-to-end test automation in the cloud.



Real Device Cloud

Test on 10,000+ real Android and iOS devices, and 3000+ browser combination cutting costs while ensuring compatibility.



HyperExecute

Accelerate testing speed by 70% with AI-Native orchestration for faster digital transformation.



Accessibility Testing

Ensure inclusive, accessible websites with TestMu AI's manual and automated Accessibility Testing tool.



Visual UI Regression

Achieve UI perfection quickly with AI-Native visual regression testing across all platforms.



+1 (866)-430-7087

www.testmu.ai

sales@testmu.ai

TEST INTELLIGENTLY. SHIP FASTER.

