Implementation of Automated Testing Solution for Voice Services

M. Ivančić, N. Štokić, T. Žilić
Hrvatski Telekom, Zagreb, Croatia
marino.ivancic@t.ht.hr
nenad.stokic@t.ht.hr
tomislav.zilic@t.ht.hr

Abstract: Implementation of automated testing solution as a support in the projects and daily operational activities for voice services is of great importance in the optimization of time and human resources. Given the growing need in the field of telecommunications for maximum availability of the service to the end-user, it is important to have a tool that can perform system control activities as well as the ability to detect errors within network elements. Hrvatski Telekom has integrated one such system into the voice core network, which has several significant advantages for engineers who will improve the quality of service they provide to their customers using such a tool. The paper describes the technical solution of the Voice Automated Testing system as well as its implementation in the live network.

Keywords: automation, testing, voice, service, implementation, integration

I. INTRODUCTION

Technology changes in the telecom network environment are becoming more and more frequent over the last years. When talking about voice core networks and voice services, major technological evolution came with the introduction of the IMS (IP Multimedia Subsystem) based on the SIP (Session Initiation Protocol) protocol both for fixed and mobile voice services. Services are becoming more complex, hardware and software updates more frequent and system lifecycles shorter. All mentioned is impacting the increase of demand for resources needed for testing and verification processes [1]. Further modernization of the telecommunication network infrastructure and migration of platforms and services to the cloud environment even intensified such processes. A logical step forward is to speed up and automate the testing and validation process and optimize available resources to the greatest extent. This paper is describing one such approach of implementing an automated testing solution for telecom voice services.

Automation as a such is a much wider area tackling all aspects of the ICT universe. With technology progressing automation tools are becoming more available, and the need for automation for any sort of process is becoming increasingly important. As the Continuous Integration/ Continuous Delivery (CI/CD) concept already become standard in the software industry, the need for automation of the testing process is of great importance, and frameworks for automated testing are being developed [2]. Test automation frameworks are a set of rules and corresponding tools that are used for building test cases [3]. They are designed to help engineering functions work more efficiently. The general rules for automation frameworks include coding standards that you can avoid manually entering, test data handling techniques and benefits, accessible storage for the derived test data results, object repositories, and additional information that might be utilized to run the tests in a suitable manner. Examples of such automation frameworks are present in telecommunication networks as well, and they are focused on the design and development of a testing framework that automates the end-to-end configuration and control of a communication testbed [4].

Choosing the right automation framework and corresponding automation testing tool is not an easy task. Implementation of the selected approach is even more demanding, taking into concern not just technical efforts but also integration into the existing processes. However, the benefits of resource optimization should exceed those efforts. This paper will present an automated testing solution for voice services in a telecommunication provider network, based on the in-house developed automation tool.

II. VOICE AUTOMATED TESTING SOLUTION

A. Need for Voice Automated Testing

As part of the project that includes the modernization of the core network elements for voice services, Hrvatski Telekom decided to implement a new solution for automating the testing of scenarios in the field of voice services to speed up and optimize the process of testing equipment, platforms, and functionalities that support

voice services, including testing mobile and fixed call scenarios. Until now, test scenarios were mostly performed physically and manually by customizing the test environment for each test scenario, which resulted in a large expenditure of time and resources. With the increasing need for more complex scenarios when testing voice services, there was a clear need to implement a new tool for voice automated testing. As already mentioned, the selection of an appropriate test automation framework and test automation tool is not a trivial task [5]. Sometimes even a lack of adequate tools can be an obstacle to test automation. High initial investment and training for tool operation are also perceived as common challenges. Anyhow, in the telecommunication voice network and services area, there are a few established suppliers that offer test automation frameworks within their automation tools and solutions. Some of the most relevant ones are Segron Automation [6], Spirent Communications [7], Sigos (Mobileum) [8], Keysight (Ixia) [9], and many others.

Even though there are quite a few off-the-shelf test automation solutions, it was decided to use a new in-house solution developed within the Deutsche Telekom group, Voice Automated Testing (VAT) tool. It is a product owned by Deutsche Telekom, developed as an internal automated testing tool for the telecommunication service providers within the group. Hrvatski Telekom was the first one to choose it and became the first telecommunications operator within the DT group to implement and integrate such a system within a live network. This paper is focused on the VAT solution description and integration in the live voice network. It will be depicted how the tool works and what possibilities the new automated system for testing voice services has, and how it was implemented in the HT environment. Further solution details in the paper are based on internal technical documentation and firsthand experience.

B. Voice Automated Testing Solution Description

The implemented solution enables end-to-end testing of voice services, which include CS (circuit switch) mobile, VoLTE (Voice over LTE), and fixed services. This includes checking registrations and call establishment, as well as additional services such as call forwarding, call waiting, call retention, conference calls, etc. To achieve the above-listed services, it was necessary to implement equipment that can support this and integrate it with the live network.

In Figure 1 it can be seen what the VAT system hardware looks like integrated within the live network. For implementation, various hardware components were needed such as attenuators for signal management, shielded boxes for controlled conditions, mobile probes containing mobile devices, IP phones, VPN router, switch, patch panel, L1 switch, SIM array, and base station. In the presented environment, three shielded boxes were used with six mobile probes inside them (Samsung S9), in addition, there were six IP phones (VoIP technology) and six USB modems (analog technology) that are located outside the shielded boxes because they do not need signal management. With the attenuator connected to the base station and WiFi access point, it was enabled to manage the signal in shielded boxes (2G, 3G, 4G, WiFi).

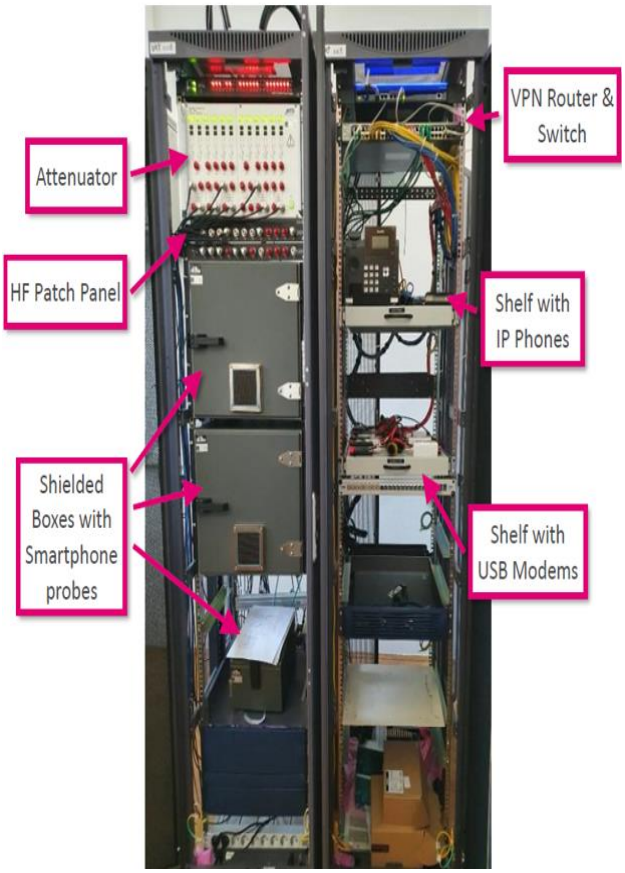

Figure 1. VAT system hardware

By integrating all these elements with live network elements, it was possible to achieve the conditions for testing different scenarios for the voice services. Users of such a system can perform different tests that can be performed multiple times without the need to set the same parameters before each test. Since the complex projects implemented in the telecommunications network are rapidly increasing, there is a need for tools that will help optimize resources in daily tasks. For the implementation of new services, regression testing, or monitoring network availability and stability (signaling, call flow, performance indicators, etc.), the VAT system should be very helpful in this area. Constant changes in the networks that are nowadays rather complex, taking into consideration cloud infrastructure lifecycle, core network applications lifecycles, new services, and features introduction, it is of vital importance to have an automated testing process to run the network appropriately.

VAT hardware equipment itself, also has its lifecycle and needs to be updated on regular basis, especially end devices like mobile phone probes which are changing very fast. In addition to that, the system should be scalable for expansion in case of new use cases implementation, new and additional probes integration, etc.

Figure 2. Shielded box with the mobile probe inside

Figure 2 shows how it looks when the shielded box is opened with a mobile probe inside (Samsung S9). VAT enables the sequential execution of several different tests individually and/or in a multi-defined block without the presence of the system user. The user of the system must be able to mark the tests to be carried out, as well as set the choice of test range that can be performed at any time of the day or night, and for each test a pass or fail result must be provided and all relevant information about the test scenario must be available. The test results together with the specified criteria are used to evaluate the test. This test may fail if the equipment test results are not good, but also if there is an error in the solution or issues in the network. After each test is performed, the solution must provide a test report - results, parameters used, test duration, etc. The report should be available in storage after the test along with screenshots of the phone during the test. It must be possible to export the network trace of the test scenario in *.pcap* format from the live network and *tcpdump* records from mobile devices. The solution must be able to test all mobile (2G, 3G, LTE), VoLTE, VoWiFi and VoIP technologies. For example, a VoLTE user calls a VoWiFi user registered on a WiFi network under a 2G signal. In addition, it can be enabled to use the feature for video recording of mobile devices during the test, which can be very helpful during the troubleshooting process. It can be configured to send email notifications to specific addresses if a series of tests fail, which can act as an alarm system. It is important to have various probes implemented to be able to perform interoperability tests within the whole network.

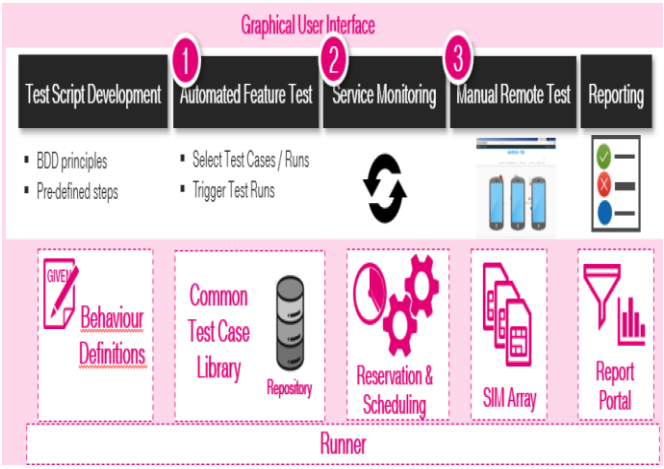

Figure 3. Automation logic

VAT could be used in several ways. Mainly it is used to automatically run a larger set of tests, but sometimes it is necessary to manually control the mobile device to achieve the desired goal. Manual control provides the same possibilities as if a mobile device is in a tester's hand. This possibility makes it easier to carry out specific test cases because it can be managed to use multiple devices at the same time and control their signal. To automatically run multiple tests within the VAT system, it is needed to have different tools to achieve this, and therefore multiple open-source platforms are used (Jenkins [10], Gitlab [11], Docker [12], ReportPortal [13]). Jenkins (automation server) is a tool used to create a "job" by selecting the probes, SIM cards and tests selected to be performed. Docker (containers) tool will use the parameters from the Jenkins "job" and will create an image with selected probes, selected SIM cards and run selected tests with them (before each test the device settings are reset so the test can be executed properly). Through this process, the system reserves selected mobile devices and connect them to selected SIM cards located in the SIM array, and then runs tests with selected parameters stored in the GitLab repository. After the test is finished, the results of the executed test scenarios and their contents are seen (*pcap* trace, *tcpdump*...) through the reporting portal as described in Figure 3. The picture visually depicts the automation logic of the VAT system that can easily be related to the test automation framework. In this way, the idea within the DT group is to have Test Automation as a Service (TAaaS) approach, and rapidly decrease manual interaction wherever applicable.

Figure 4 shows the execution of the test on mobile devices that can be monitored in real-time along with all the information about the progress of the test execution. The average time of performing one test is about 3-5 minutes depending on the test, so depending on the number of free probes and the execution time of each test, in one hour the system can perform 10-50 tests in a fully automated mode. If compared to the manual physical testing where it needed about 5 minutes to prepare the environment for the test scenario, execute the test and

collect the traces. The biggest advantage of a tool like VAT is that it can perform test scenarios execution without human interaction and testing can be scheduled anytime including during non-working hours.

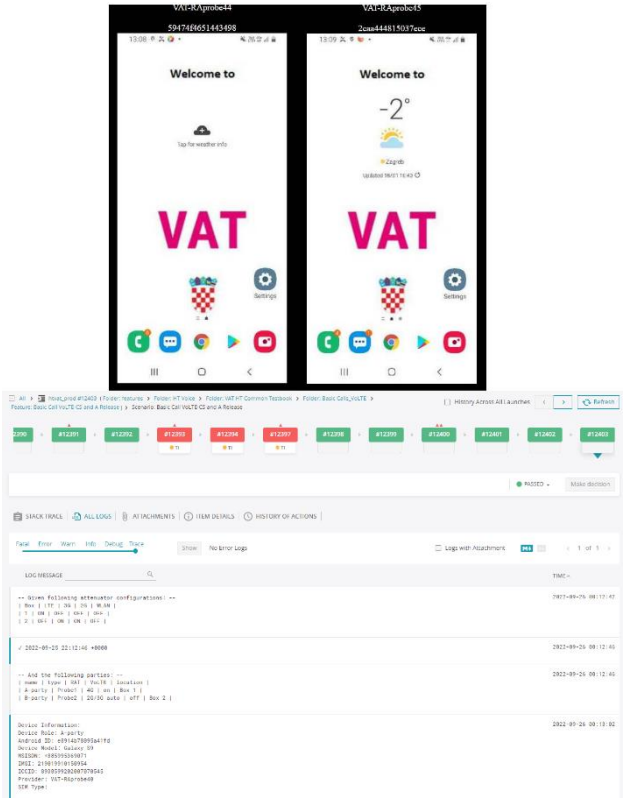

Figure 4. VAT mobile probes and ReportingPortal overview

III. Use Cases

VAT system could be used for various voice service testing scenarios like CS mobile voice services, VoLTE mobile services, fixed voice services, and should cover use cases such as:

- Availability and basic calls (registration, call establishment)
- Calling Line Identity Presentation/Restriction
- Additional services (call forwarding, call waiting, call hold, conference calls)
- HD Voice (VoLTE)
- Long-duration calls
- Interactive voice response (announcements)
- Ringtone sound detection
- SRVCC testing (loss of LTE signal during the call)
- SMS testing
- VoWiFi testing
- Inbound roaming testing
- Interoperability between CS mobile, VoLTE and fixed voice services

To provide a complete overview of calls from one end to the other, the system collects traces from the network (*pcap*) and network logs from mobile devices (*tcpdump*) if the user needs them for a more detailed analysis. The system should have full control over the devices so that it is possible to run test scenarios for voice services (registration, call establishment and call control) on them. For the devices to know exactly what they need to do, and what steps they must complete to execute a certain scenario, they use written commands that are found within the test scenarios. The test scenarios are written in the Gherkin programming language [14], which is convenient for automation because it uses plain English text that is understandable to almost anyone. For every step (command) written in Gherkin, there is a back-end code written in Ruby programming language [15].

```
@bcvolte001
Feature: Basic Call VoLTE-VoLTE and A Release

  Background:
    Given following attenuator configurations:
      | Box | LTE | 3G  | 2G  | WLAN |
      | 3   | ON  | OFF | OFF | OFF  |

    And the following parties:
      | name    | type   | RAT | VoLTE | location |
      | A-party | Probe5 | 4G  | on    | Box 3    |
      | B-party | Probe6 | 4G  | on    | Box 3    |

  Scenario: Basic Call VoLTE-VoLTE and A Release
    When A-party calls B-party
    And A-party verifies that the call state is 'dialing'
    And A-party verifies that the call state is 'alerting'
    And B-party verifies that the call state is 'incoming'
    And B-party verifies that the call is 'ringing'
    And A-party number is displayed at B-party
    Then B-party answers
    When B-party verifies that the call is 'established'
    Then A-party and B-party hear each other
    And all parties verify that they are camped on '4G'
    Then after 35 seconds A-party hangs up and becomes 'idle'
    And B-party verifies that the call state is 'idle'
```

Figure 5. Example of test in Gherkin

Figure 5 shows the test scenario under the unique tag @bcvolte001 with the configured signal settings inside the shielded box 3 containing probe 5 as the A-party and probe 6 as the B-party. A-party and B-party are VoLTE users, and A-party makes a 4G (VoLTE) call to B-party, which is also a VoLTE user. In the test, there are steps to verify call establishment where B-party answers the call. When the media check is verified on both sides during the call, it needs 35 seconds to pass so that the A-party ends the call.

Within this concept, a variety of voice service scenarios can be covered, including the most complex tests, and all the activities can be predefined and scheduled in advance, so no human interaction is needed while the

test is executed. As already stated, the VAT system uses a repository that contains most of the fixed and mobile voice services use cases present in the telecommunication network. It is up to the VAT system administrators and end devices availability how many use cases will be automated.

IV. Further Development

So far it was explained how the VAT system can help in daily operations, but there are still aspects of the solution that could be improved to make the system even better. The first point is to continue the development of the existing functionality and make the system even more robust, accurate, optimized and faster in task execution. Further on, support for the various types of end devices is needed as well, to be adaptable for any network environment. For example, at first only Android end devices were supported but now iOS support was introduced as well.

The next evolution step would be an introduction of some sort of AI (artificial intelligence) concept that would move automation to an even higher level. In that matter, to make the existing solution to be as high-quality as possible the results of the performed tests that are received in *.pcap* format should be verified automatically, so the system can decide whether the signalization in the call flow for a certain test is good or not (pass/failed). The execution of the test itself can be done successfully, it can look like the functional execution of the test scenario seems good, but it can happen that certain call signaling part is not done as expected. Because of such scenarios, in the current solution, human effort is needed to analyze the call flow for a specific use case in more detail to be able to confirm whether the performed test scenario is correct or not.

Another possible way forward is expanding the scope of the existing system which was intended for voice services only. With the introduction of the 5G network, there is a need to cover 5G-related voice services [16] but also the data network services as well. With additional functionalities for Data Automated Testing (DAT), automated processes to test data services will be implemented. Within the same concept, only different tests written in the same language would have the ability to test scenarios such as surfing the web, downloading, uploading, and measuring the performance indexes.

The solution has been recognized within the Deutsche Telekom group, and other national companies that are part of the group are starting to implement VAT solution in their networks. Even though this is an in-group (in-house) approach, it is still a commercial product that is offered to the group members in form of a shared services business model [17].

V. Conclusion

There is no doubt that automation in general is becoming an inevitable part of any ICT system introduction and maintenance, and telecommunication networks are no different in that matter. From all the above it can be concluded that the implementation of such a solution is fully justified in terms of time optimization and human resources. Furthermore, while continuously using the VAT system it can detect anomalies in the network if there is an active task where the same tests will be always performed. The system can also be set up to be used by non-technical people who can extract useful information for activities that are important to the company for some analysis and finding new business models that can be offered to the end user. The system is of great value for network and service regression testing when network changes, like system upgrades or new functionality introduction, are done and is very important to check and verify if everything is working as it should be. Automation concept evolution is not stopping there, with new use cases, scope enhancement, and new functionalities being introduced, this approach should have a bright future ahead.

Literature

[1] D. Gonçalves, A. Amaral, A. Costa, P. Sousa, “Towards automated test and validation of SIP solutions”, Telecommunication Systems, 61, 2016, pp. 579-590

[2] T. Fehlmann, E. Kranich, "A framework for automated testing", Systems, Software and Services Process Improvement: 27th European Conference, EuroSPI 2020, Düsseldorf, Germany, September 9–11, 2020, Proceedings 27. Springer International Publishing, 2020., pp. 275-288

[3] G. Hegde, A. Badkar, “Popular Test Automation Frameworks”, BowserStack web and mobile testing platform, 2020, retrieved from: https://www.browserstack.com/guide/best-test-automation-frameworks

[4] A. Díaz Zayas, B. García García, P. Merino, “An end-to-end automation framework for mobile network testbeds”, Mobile Information Systems, 2019.

[5] P. Raulamo-Jurvanen, M. Mäntylä, V. Garousi. "Choosing the right test automation tool: a grey literature review of practitioner sources", Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering, 2017, pp. 21-30

[6] Segron Automation Website, retrieved from: https://segron.com/

[7] Spirent Communications Website, retrieved from: https://www.spirent.com/

[8] Mobileum (Sigos) Website, retrieved from: https://www.mobileum.com/

[9] Keysight (Ixia) Website, retrieved from: https://www.keysight.com/us/en/cmp/2020/network-visibility-network-test.html

[10] Jenkins Website, retrieved from: https://www.jenkins.io/doc/

[11] Gitlab Website, retrieved from: https://docs.gitlab.com

[12] Docker Website, retrieved from: https://docs.docker.com

[13] ReportPortal Website, retrieved from: https://reportportal.io/docs

[14] Cucumber (Gherkin programming language) Website, retrieved from: https://cucumber.io/docs/gherkin/

[15] Ruby (programming language) Website, retrieved from: https://www.ruby-lang.org/en/

[16] Z. Zhao, X. Liu, J. Miao, S. Zhou, X. Cheng, L. Xu, S. Liu, “Research on Automated Test and Analysis System for 5G VoNR Service”, Signal and Information Processing, Networking and Computers: Proceedings of the 8th International Conference on Signal and Information Processing, Networking and Computers (ICSINC), Singapore: Springer Nature Singapore, 2022, pp. 1396-1403

[17] T. Žilić, "Shared Services Business Model in ICT environment," 2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO), Opatija, Croatia, 2020, pp. 1367-1372