IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Blend of Manual and Automation Testing Approach Towards Finer Software Quality

[1]Prem Kumar, [2] Dr. R. Nagaraja
[1]Mtech Student, [2] Professor, PG & Research Coordinator
[1]Information Science and Engineering,
[1]Bangalore Institute of Technology, Bangalore, India

Abstract: A better software quality is always desired when a software is developed and provided to the users. Especially this becomes a matter of great concern when the user is provided with a user interface to perform operations and interact with the software functionality. To ensure that the software quality of such delivered solutions is high and efficient, a great approach is to perform the thorough check of the software functionality through a manual as well as through automation. The importance of the software testing involving both the manual and automation testing becomes paramount as the number of features and tests to be performed grows exponentially with every new release. In this work, the manual and automation testing approach is followed, and results are noted down. The results show a considerable amount of advantages when coming to any single form of testing.

***Index Terms* – Manual testing, Automation testing, malware analysis, data configuration.**

I. INTRODUCTION

It is of utmost importance that the software quality and functionality is thoroughly tested before the software is developed and provided to the user for further use. Specially, if the software is time and performance intensive, it requires a lot of efforts to stay up to the mark ensuring that the user is provided with highest quality of the software. Each functionality needs to be tested and the results should be noted down. Once these testing results are noted, the changes can be further done which can be then followed by the enhancements to be done in the software that is going to be presented to the customers. The manual testing process brings up challenges in the number of man hours required for performing the tests and other configurations that needs to be done for every new test. Even the simultaneous checking of these test scenarios in different set of environments. In this corner of the testing challenges, it can be combined with the automation process of testing the software functionality which further improves the software quality. Even with the automation, graphical user interface poses significant challenges [1]. These challenges can be of being in sync with all other tests that are required to be run one after the other and it increases when the tests performed are interdependent on each other which leads to execution delays if the first test scenario is delayed. If the tests are time dependent, it adds more to the testing challenges. Such challenges require that the testing of a software quality and functionality is done through a combination of manual and automation test approach. In the RSA NetWitness software, the software quality is of utmost importance and it requires that each of the functionality is properly working and as expected even when the load increases or when it is deployed in different platforms under different configurations. Such testing is deployed in the context of malware analysis component and ensured that it provides the best of results when delivered to the customer.

II. LITERATURE SURVEY

It has been read and observed in different research work done in the field of testing and moreover with the help of automation. Both the manual and automation test approach brings with it the new set of challenges and scenarios that are required to be considered before forming any strategy to take over the task of testing software quality. Mostly, the automation test framework that is utilized and in top of other frameworks is the selenium testing framework [2]. It helps in automation of the test cases and running the same by scripts. Unless and until the scripts are used, the execution of the automated test remains a big challenge. Similar challenges have been listed in various other research pertaining to manual and automation approach. The only manual approach to be followed is never optimum in terms of costs, quality or the quantity of test scenarios that are covered of the software functionality. RSA NetWitness software is the one that forms an enhanced tool to have a deeper insight into the organization network and ensure that each and every packet or log that is flowing in and out of the organization is checked and an alert is raised if the transfer of such packets or logs seems to be suspicious. Similarly, malware analysis component of the RSA NetWitness software ensures that files such as PDF, word or other executables files received by the organization network and flowing to the endpoint machine is monitored. If any file is found to be suspicious or not fit for use, it is marked as malicious and the user is stopped from accessing such files preventing any further damage or loss that could be occurred by accessing the file. However, there are different ways in which these scans are run by this component. The three different approach this component follows to ensure that the file is safe to use are by continuously polling the data from another device that has indexed data along with the

files received and performing scans only for the files that are found suspicious. The second scan approach is to only scan the files that a user of the software request to scan and provide the results. Last approach is to manually upload the files of interest to the malware analysis component and check if the file is suspicious or fit for use for further purpose in the organization. These different approaches provided by the malware analysis software provides the user with flexibility to use the functionality as they desire to.

III. PROPOSED SYSTEM

The proposed system is the one that includes the best practices of both the manual and automation test approach. In this, it is ensured that benefits of both are gradually taken and deployed for successful testing of the software functionality and improving the quality of the software at the same time. Figure 1 given below defines the process of considering the manual and automation test approach for the malware analysis component of the software.

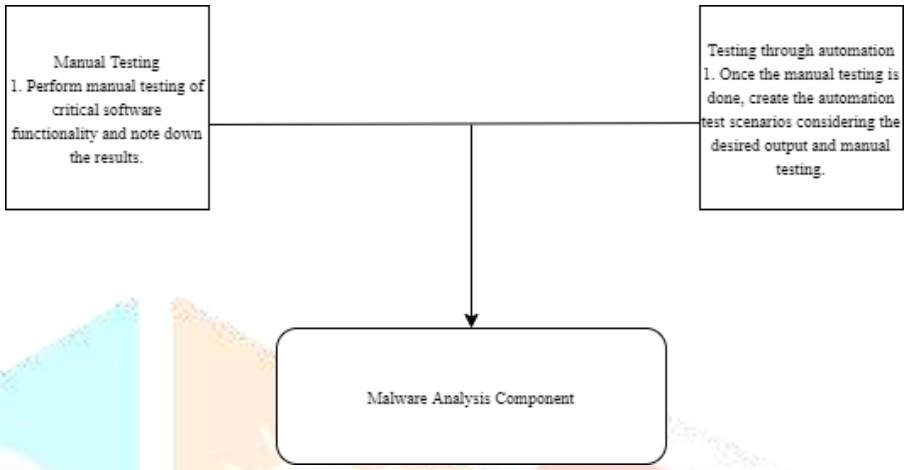

Figure1: Combination of manual and automation testing

In the figure 1, it only specifies that the automation the process of manually performed tests for the critical functionality of the software. However, this is not only in the proposed system, it involves the configuration and automation of other test scenarios that will then cover a broader part of the software with the user interface testing too. First part that is to be considered here is results of the manual testing of the functionality provided by the malware analysis component. Once the manual testing is done through the user interface, it is to be considered as a part of the automation test framework and to be added in the further tests that should be performed simultaneously on the software deployed on different platforms in the customers environment. Automation test scenarios should also consider the configuration settings that can be done by a customer. Such configuration should also be done though automation once the manual testing is performed and the desired outcome is noted.

When the tests performed through automation fails due to any reason that may arise in the environment, it is essential to bring back the process of manual automation immediately to find if the software is actually having the necessity to be fixed or it is caused by the flakiness of the environment in which the test is run or due to the automation framework itself that might need to be fixed. Hence, combining the manual and automation test approach brings us to a point where the software functionality is checked in all cases and quality of the software is ensured to provide to the best results when put in the customer use. This testing framework is to be deployed using the selenium and other data configuration framework that will be solely used for the purpose of malware analysis component configurations.

IV. METHODOLOGY

Malware analysis component of the RSA NetWitness software interacts with the other services that has the data indexed and kept in a format that can be queried. The device that indexes the data after the logs and packets are captured in the RSA NetWitness software is the concentrator [3]. The concentrator device has the capability to mark the files as suspicious based on some predefined parsers that are used. Hence, the data can be collected in case of continuous mode of scanning from the concentrator and further scans by the malware analysis service can be performed. Also, this component deals with the interface where the user can manually upload the files to be scanned. Such interactions are to be carefully designed and configured when the testing is done through automation. To ensure that the automation testing is done as expected, the manual testing is first performed. Once done, the automation test framework should also consider the user interface scenarios where the elements might not be displayed properly due to network or any such issues. Concentrator component collects the data from devices that do the work of collection of the logs and packets continuously through the designated interfaces. The packets can be captured in a promiscuous mode where each packet flowing through the organization network is captured and stored in the encrypted format back to the storage device. In a similar way, the logs are collected form different event sources. Even the logs from the computers used in an organization are collected. These packets and logs do not form any meaningful information unless they are properly formatted using the logic in concentrator. Once such logical operations are performed, these devices are ready to communicate with each other and perform a task that is of utmost importance to any organization in the pursuit of complete security. RSA NetWitness malware analysis component further communicates with the concentrator to identify and detect the threats posed by the files used [4]. The interaction between these components is of great importance that form the useful information. However, the functionality of these components is separated due to the load factor they may face at the time of high throughput.

V. RESULTS AND DISCUSSION

Proposed system is implemented and deployed using the different tool and technologies. The framework used for configurations is called as the data config tool and the framework used for test scenarios executions is the one based on selenium testing tool. These frameworks are used to then execute different tests with different configurations and ensure that the software functionality and quality is improved. Before these tests are run through automation, the manual testing is performed that entails more of human interaction with the machine and user interface of the software. In this scenario, testing can be performed in a way that cannot be always handled through automation. When the test cases are noted post this process, the configuration framework and selenium tool is used to design the test cases which are then executed. The test results after these processes show the drastic reduction in the time testing is performed in different platforms and configurations. Also, the complete malware analysis component functionality is covered again ensuring the quality of the software is never negotiated. Figure 2 and 3 shows the execution of data configurations and tests that are written using the selenium framework.

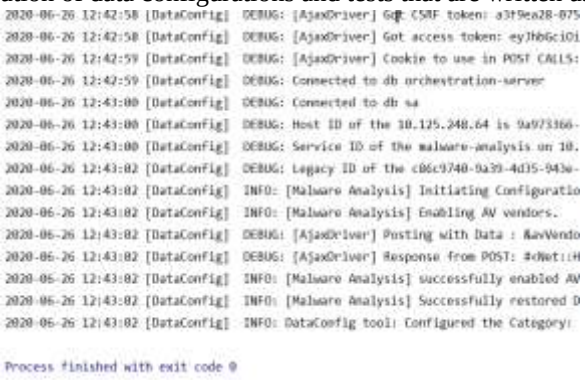

Figure 2: Data Configuration Execution for Malware Analysis

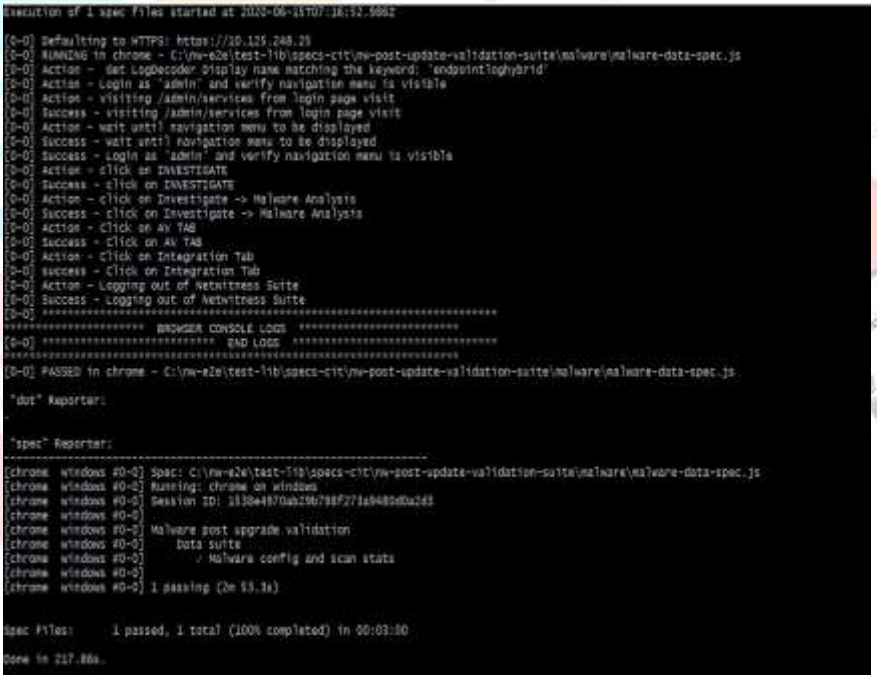

Figure 3: Test results execution using selenium framework

These figures displayed here provide us with the configurations and test that were performed one after the other. These include the manual testing of the software functionality too which combines the best of both the test approaches and improves software quality.

VI. CONCLUSION

The conclusion after execution of test and configuration scenarios done after the manual and automation approach is that combining the best of both cases presents us with a powerful mechanism to ensure the quality of the software provided to the customers is efficient and thoroughly tested. Both the manual and automation approach together also leads to formation of various test scenarios that cannot be taken up simply by considering the functionality of the software that it is intended to provide. Hence, combining both the manual and automation approach of testing provides us a way better performance in terms of test execution and formation of new test cases. It also ensures that all different types of test scenarios are executed well in advance in different customer simulated environments.

REFERENCES

[1] P. Aho and T. Vos, "Challenges in Automated Testing Through Graphical User Interface," 2018 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW), Vasteras, 2018, pp. 118-121, doi: 10.1109/ICSTW.2018.00038.
[2] P. Ramya, V. Sindhura and P. V. Sagar, "Testing using selenium web driver," 2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT), Coimbatore, 2017, pp. 1-7, doi: 10.1109/ICECCT.2017.8117878.
[3] RSA LINK, Getting Started with NetWitness Platform. [Online]. Available from: https://community.rsa.com/docs/DOC-81171 [Accessed 4th February 2020].
[4] RSA, Threat Detection and Response. [Online]. Available from: https://www.rsa.com/en-us/products/threat-detection-response [Accessed 10th February 2020].