Automatic Software Quality Assurance

Laura Ureche
Faculty of Automatic Control and Computer Science
“POLITEHNICA” University of Bucharest
laura.ureche@gmail.com

1. Introduction
The quality of a software product is achieved by merging development and testing until they became indistinguishable.

Software products and their environments change continuously; they should be corrected, enhanced and ported to new platforms.

The main reason for which software engineers perform automated testing is to ensure the quality of the modified systems. Once automated tests are created they can easily be repeated and they can be executed to perform tasks impossible with manual testing. The tests compare the results to the expected behaviour and report the success or failure.

The performance testing is used to determine how a product works in terms of responsiveness and stability under particular conditions. Also, it measures, validates and verifies other quality attributes, such as scalability, reliability and resource usage.

2. Aim
The purpose of this paper is to introduce a measure for the quality of a specific software product by analysing the results obtained from the performance testing.

The obtained results and the tests weight are added into a grade grid used to reach decision and quality measure; representing the novelty in the testing area.

The software product chosen for this paper is Avira ICAP. Avira ICAP provides reliable protection by scanning, content filtering and, if necessary, blocks access to all files coming from or going to the Internet. This product makes use of the detection capabilities of Avira engine, adding support for the ICAP protocol, enabling the use of Avira’s malware detection technology in conjunction with other ICAP clients.

The aim of the tests is to determine the scanning speed and resource consumption for Avira ICAP, when scanning various types of files, with various number of clients. Separate scenarios were executed with or without content filtering.

3. Method
All tests that will further be presented were executed on Intel Core i7 Quad Core, with 16GB of RAM for the machines running Avira ICAP and the test script and on a separate machine for the File Server. All machines were entirely dedicated to testing. The tests were executed using SuSe Linux Enterprise Server 11 SP2 operating system on the machine running Avira ICAP.

The results that will be presented have been gathered from tests with or without content filtering. All tests mentioned above are executed with a variable number of workers.

For the performance testing of Avira ICAP the sample file set consists of browsing files.

4. Results
The results section will contain two parts. The first part shows scanning speed results for each of the two executed test scenarios. The second part contains details about the resources loading, as they were measured while executing the test scenarios.

Visual representations of the scan speed evolution are illustrated. The graphs that will be presented will illustrate the link between the number of workers and the scanning speed.

Visual representations of the memory and disk space loading are also shown. The graphics illustrate the relation between the number of workers and the consumed resources in terms of reserved memory and disk space used for temporary files.

5. Conclusions
Considering the results obtained from the performance testing, the selected quality aspects can be mentioned:

• Memory consumption is proportional with the number of workers;
• The spread between maximum and minimum reserved memory amount is approximately constant;
• Scalability, the software product serves big number of workers in a short time;
• Multiple instances on different listen interfaces can be installed if the number of workers increases;
• Reliability, during the scan of different files may appear some errors if the limit system is reached or the system is out of disk space;
• The malware detection rate of Avira engine is 100% according to the AV-Comparatives from November 2013.

Using (generated) automated tests led to more issues found in a shorter time. In this way, further costs for fixing the problems could be avoided.

6. Acknowledgement
Many thanks for their support to Avira Operations GmbH & Co. KG, AV-Comparatives and Prof. Ing. Simona Caramihai, Ph.D “POLITEHNICA” University of Bucharest.

7. References